ADAPTIVE PREFERENCE LEARNING WITH BANDIT FEEDBACK: INFORMATION FILTERING, DUELING BANDITS AND INCENTIVIZING EXPLORATION

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Bangrui Chen

December 2017

ADAPTIVE PREFERENCE LEARNING WITH BANDIT FEEDBACK: INFORMATION FILTERING, DUELING BANDITS AND INCENTIVIZING EXPLORATION

Bangrui Chen, Ph.D.

Cornell University 2017

In this thesis, we study adaptive preference learning, in which a machine learning system learns users' preferences from feedback while simultaneously using these learned preferences to help them find preferred items. We study three different types of user feedback in three application setting: cardinal feedback with application in information filtering systems, ordinal feedback with application in personalized content recommender systems, and attribute feedback with application in review aggregators. We connect these settings respectively to existing work on classical multi-armed bandits, dueling bandits, and incentivizing exploration. For each type of feedback and application setting, we provide an algorithm and a theoretical analysis bounding its regret. We demonstrate through numerical experiments that our algorithms outperform existing benchmarks.

BIOGRAPHICAL SKETCH

Bangrui Chen was born on June 24, 1990. He grew up in Qingdao and has been interested in solving math puzzles since he was a kid. Before he came to the United States, he received a Bachelor of Science in Mathematics from Nanjing University, China.

In Fall 2013, he came to Cornell to pursue a Ph.D. degree in the School of Operations Research and Information Engineering, with a concentration in Applied Probability and Statistics. He studied under the supervision of Professor Peter Frazier and his research focused on the exploration vs. exploitation trade-off in adaptive preference learning problems.

In his spare time, he enjoys tennis, poker, and video games.

To my family.

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my Ph.D. advisor Peter Frazier for his unconditional support during my doctoral research. His immense knowledge in the field, enthusiasm about research and unparalleled creativity never ceased to amaze me. I could not finish my research without his unreserved help. Besides research, his superb time management ability and effective communication skills set an excellent example for mentors. He is always there when I need his help and I am extremely fortunate to have Peter as my doctoral advisor.

I would also like to thank Thorsten Joachims and Huseyin Topalogu for serving on my dissertation committee and providing constructive feedback for my research and my thesis. My sincere gratitude also goes to the faculty of Operations Research and Information Engineering. I am grateful that I was able to attend many interesting courses taught by world-renowned researchers, who are always willing to discuss and share their thoughts.

To my friends Wei Chen, Jing Xie, Jiayi Guo, Chek Hin Choi, Guo Yu, Zhengdi Shen, Jiekun Feng, Ze Jin, Zi Ye, Yuan Cheng, Jialei Wang, Pu Yang, Yuhang Ma, Tiandong Wang, Tom Fei, Weilong Guo, Massey Cashore and many others, thank you for making my experience at Cornell unforgettable. I will always remember the stimulating discussions we had and the fun time we shared.

During my time at Cornell, I had the rare opportunity to try out five different internship positions and explored my interests. I would like to thank my intern managers, Xiaofei Liu, Ricky Shi, Steven Oven, Chris Voekler and Pinxing Ye, who provided me the opportunities to work on interesting real-world problems and brought me to the world of quantitative finance. I would also like to thank

my friends who I met during my internship search, Keven Li, Weifeng Cheng, Kewei Ming, Chanjuan Pan, Pengfei Yang, Qiran Wang, Weichen Wang and many others for their valuable career advises and thoughts about quant finance.

Last but not the least, I would like to thank my grandma, my parents, my wife and all of my family members for their support, sacrifice and encouragement in my life. They always believe in me even at the times I do not believe in myself. I could not finish this thesis without their constant encouragement.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Preference learning (Fürnkranz & Hüllermeier, 2010; Busa-Fekete & Hüllermeier, 2014) is a subfield in machine learning, in which a machine learning system tries to create a supervised learning model that correctly predicts a human's preferences over a collection of items, based on feedback. Typically the human is a user of a website or mobile app. For instance, Netflix wants to learn a user's preference based on his/her past watching history so that it can make a better movie recommendation; E-commence websites such as Amazon want to learn a user's preference to identify that user's favorite brand or product.

Traditionally, such a system would infer a user's preference through a regression model trained on relevance feedback from past interactions (Agarwal et al., 2011a,b). However, when new users come to the system, or when item contents or user interests change, sufficient training data may not be available. In such "cold-start" situations (Schein et al., 2002), it becomes attractive to pursue *adaptive* preference learning, in which the system would interact with users in a way that both provides them items that they prefer, but also solicits feedback that is likely to be useful in learning their preferences. In doing so, we must trade the benefits of exploring a user preferences against possible degradation over the short term in the quality of the user's experience. This is an example of the so-called exploration vs. exploitation tradeoff (Auer, 2002; Sutton & Barto, 1998).

In this thesis, we study adaptive preference learning with bandit feedback (Dani et al., 2008). More specifically, we consider three different types of

user feedback and we study each type of user feedback with a specific application under the adaptive preference learning framework. First, we study cardinal feedback (Ailon et al., 2014), in which a user provides a real number representing a rating or reward for the satisfaction produced by an item. We study cardinal feedback with application in information filtering systems in Chapter 2. Second, we study ordinal feedback (Ailon et al., 2014), in which a user provides a binary value representing which of two presented items the user prefers. We study ordinal feedback with application in personalized recommender systems in Chapter 3 and Chapter 4. Third, we study attribute feedback, in which we observe a vector of real numbers representing user ratings for each of several attributes of an item, but may not observe the user's overall level of satisfaction. For example, a review aggregator such as Yelp or Tripadvisor may ask its users to review a restaurant for its location, food, price, service etc. We study attribute feedback with application in review aggregators in Chapter 5. In all three applications, we propose algorithms that have both theoretical performance guarantees and strong empirical performance.

In adaptive preference learning, we face a tradeoff between providing items that our current model believes the user prefers (exploiting our current model) vs. asking for feedback about items about which our model has significant uncertainty (exploring). This exploration vs. exploitation tradeoff has been studied heavily in the context of the multi-armed bandit problem (Robbins, 1952), and we leverage ideas from this field in our work on adaptive preference learning. In the multi-armed bandit problem, a gambler must decide which slot machine to pull at each time t so that he can maximize his cumulative reward. In this process, the gambler hopes to learn each slot machine's reward as soon as possible, while also pulling engaging slot machines that are known to have

high payoffs.

As an example of the connection between multi-armed bandits and adaptive preference learning, learning a user's preference in an information filtering system can be seen as a special case of the Bayesian contextual linear multi-armed bandit problem (Agrawal & Goyal, 2013; Chu et al., 2011; May et al., 2012; Cesa-Bianchi & Kakade, 2011). The context is the feature vector for the item, and two arms are available: pulling the first arm corresponds to forwarding the item, and the user provides a feedback corresponds to the item's relevance; pulling the second arm corresponds to discarding the item, and has 0 as the feedback.

Dueling bandits are another variation of the multi-armed bandit problem that are closely related to adaptive preference learning. In the dueling bandits problem, we are faced with a collection of arms, and pull a pair of arms while observing noisy binary feedback indicating which arm is better for each pulled pair. As in the classical multi-armed bandit problem, we wish to pull arms to quickly learn which arm is the best while minimizing the number of pulls of suboptimal arms. Dueling bandits were introduced by (Yue & Joachims, 2009), motivated by interactive optimization of web search and other information retrieval systems. The advantage of the dueling bandits formulation over the classical multi-armed bandits formulation in this application setting is that pairwise comparison results can be reliably inferred from implicit feedback, for example through interleaved rankings in Radlinski et al. (2008), in contrast with cardinal evaluation obtained from explicit feedback, which is typically difficult to obtain, biased, and requires careful calibration (Joachims et al., 2007; Yue et al., 2012).

Another variation of the multi-armed bandit problem that has attracted attention recently is called incentivizing exploration (Frazier et al., 2014; Han et al., 2015). In this thesis, we explore incentivizing exploration for attribute feedback in adaptive preference learning. In incentivizing exploration, instead of learning the user's preferences, the principal (or the system) wants to explore the features of each item based on feedback from the myopic users (Mansour et al., 2015). Without any incentives, an item/product that seems worse initially may remain unexplored by myopic agents. In this problem setting, we as the principal wish to design an incentivization strategy to maximize the cumulative social welfare of a sequence of myopic users within a reasonable payment budget.

1.1 Thesis Organization

In Chapter 2, we study cardinal feedback with application in information filtering systems. We present a Bayesian sequential decision-making formulation of the information filtering problem, in which an algorithm presents items (news articles, scientific papers, tweets) arriving in a stream, and learns relevance from user feedback on presented items. We model user preferences using a Bayesian linear model, similar in spirit to a Bayesian linear bandit. We compute a computational upper bound on the value of the optimal policy, which allows computing an optimality gap for implementable policies. We then use this analysis as motivation in introducing a pair of new Decompose-Then-Decide (DTD) heuristic policies, DTD-Dynamic-Programming (DTD-DP) and DTD-Upper-Confidence-Bound (DTD-UCB). We compare DTD-DP and DTD-UCB against several benchmarks

on real and simulated data, demonstrating significant improvement, and show that the achieved performance is close to the upper bound.

In Chapter 3, we study ordinal feedback with application in personalized recommender systems. We consider online content recommendation with implicit feedback through pairwise comparisons, formalized as the so-called dueling bandit problem. We study the dueling bandit problem in the Condorcet winner setting and consider two notions of regret: the more well-studied strong regret, which is 0 only when both arms pulled are the Condorcet winner; and the less well-studied weak regret, which is 0 if either arm pulled is the Condorcet winner. We propose a new algorithm for this problem, *Winner Stays* (WS), with variations for each kind of regret: WS for weak regret (WS-W) has expected cumulative weak regret that is $O(N^2)$, and $O(N \log(N))$ if arms have a total order; WS for strong regret (WS-S) has expected cumulative strong regret of $O(N^2 + N \log(T))$, and $O(N \log(N) + N \log(T))$ if arms have a total order. WS-W is the first dueling bandit algorithm with weak regret that is constant in time. WS is simple to compute, even for problems with many arms, and we demonstrate through numerical experiments on simulated and real data that WS has significantly smaller regret than existing algorithms in both the weak- and strong-regret settings.

In Chapter 4, we continue our focus on ordinal feedback and study dueling bandits with weak utility-based regret when preferences over arms have a total order and carry observable feature vectors. The order is assumed to be determined by these feature vectors, an unknown preference vector, and a known utility function. This structure introduces dependence between preferences for pairs of arms and allows learning about the preference over one

pair of arms from the preference over another pair of arms. We propose an algorithm for this setting called *Comparing The Best* (CTB), which we show has constant expected cumulative weak utility-based regret. We provide a Bayesian interpretation for CTB, an implementation appropriate for a small number of arms, and an alternate implementation for many arms that can be used when the input parameters satisfy a decomposability condition. We demonstrate through numerical experiments that CTB with appropriate input parameters outperforms all benchmarks considered.

In Chapter 5, we study attribute feedback with application to maximizing the aggregate social welfare of short-sighted consumers using review aggregators such as Yelp and Amazon. In this setting, arms have unknown multivariate attributes, and agents have heterogeneous utility functions that map these attribute vectors onto utilities. All agents see noisy observations of the attributes of arms pulled by previous agents. We propose a simple policy that usually exploits, and incentivizes exploration only when an arm would be pulled with probability below a time-varying threshold by unincentivized myopic agents given the current posterior. With the assumption that each arm is some agents' best arm, we prove the cumulative expected payment is bounded by $O(N^2)$ and the cumulative expected regret is bounded above by $O(N^2 + M(\log(T))^2)$, where M is an upper bound on the limiting marginal density of those agents who are almost indifferent between their best and second best arms. Our results show that heterogeneity in preferences can provide "free exploration," reducing regret as compared to the single-preference unincentivized setting.

CHAPTER 2

THE BAYESIAN LINEAR INFORMATION FILTERING PROBLEM

2.1 INTRODUCTION

In this Chapter, we study cardinal feedback with application in information filtering systems. Information filtering systems automatically distinguish relevant from irrelevant items (emails, news articles, intelligence information) in large information streams (Foltz & Dumais, 1992). We present a Bayesian sequential decision-making formulation of this problem, where user interests are described by a Bayesian linear model, similar in spirit to a Bayesian linear bandit (Agrawal & Goyal, 2013). The first contribution of our work is to construct an instance-specific computational upper bound on the value of a Bayes-optimal strategy, which may be used to bound the optimality gap for implementable heuristic policies. Our upper bound is most naturally applied to items whose features are weights from a topic model (Blei & Lafferty, 2009) or other mixture model, but can also be applied to other linear models. Our second contribution is to use the idea of decomposing the problem into a collection of forwarding problems with one-dimensional feature "vectors", developed in the construction of the upper bound, to create a pair of heuristic policies, jointly given the name *Decompose-Then-Decide (DTD)*. The first heuristic, called DTD-Dynamic-Programming (DTD-DP), solves each one-dimensional forwarding problem using stochastic dynamic programming, while the second, called DTD-Upper-Confidence-Bound (DTD-UCB), uses the upper confidence bound policy with a learning parameter that is adjusted based on the distribution of feature vectors in the given direction. Finally, we evaluate

our upper bound and proposed policies on real and simulated data, and find that our upper bound is typically tight, and that DTD-UCB outperforms a number of benchmarks, including UCB and Linear Thompson Sampling, in all problem instances.

The traditional approach to adaptive information filtering trains on historical feedback and does not actively explore to get the most useful feedback. However, there has been some work on active exploration in information filtering. Zhang et al. (2003) studies a Bayesian decision-theoretic version of this problem in which a univariate score is observed for each item, and relevance is related to this score via logistic regression. The system does active exploration by valuing the information that results from forwarding, via a one-step lookahead calculation. The multi-step Bayes-optimal policy is not calculated or characterized. Zhao & Frazier (2014) studies another Bayesian decision-theoretic version of this problem in which items are described by a hard clustering scheme, and users have independent heterogeneous preferences for item clusters. A computational procedure for calculating the (multi-step) Bayes-optimal policy is provided. However, the learning scheme used does not allow learning user interest in one category from interactions with other related categories, making it difficult to scale to fine-grained item representations.

A much larger literature on active exploration may be found in work on the multi-armed bandit problem (Robbins, 1985). Indeed, the information filtering problem we study can be seen as a special case of the (Bayesian) contextual linear multi-armed bandit problem (Agrawal & Goyal, 2013; Chu et al., 2011; May et al., 2012; Cesa-Bianchi & Kakade, 2011). The context is the feature vector for the arriving paper, and two arms are available: pulling the first arm

corresponds to forwarding the paper, and provides a reward corresponding to the paper's relevance, minus some cost for the user's time; pulling the second arm corresponds to discarding the paper, and has known value 0.

While much of the work on multi-armed bandits, including work specifically on linear and contextual bandits, has focused on asymptotic regret guarantees when latent parameters (in our case, the vector of user preferences for features) are chosen by an adversary, we focus on the Bayesian setting, where we assume that latent parameters are drawn from a prior probability distribution.

Our assumption of a Bayesian framework has advantages and disadvantages. The main advantage is that it supports good performance when the amount of feedback received is small (of great importance in the cold-start setting). In contrast, algorithms designed to have regret with an optimal rate in the linear bandit setting, such as the PEGE algorithm in Rusmevichientong & Tsitsiklis (2010), typically need a number of interactions at least as large as the dimension of the feature vector, which may be hundreds of dimensions or more. A Bayesian algorithm can do well much sooner than this, by using information embedded in the prior that, for example, most users have little preference for a particular feature, or that users who prefer one feature tend to not prefer another feature.

The main disadvantage of the Bayesian framework is that choosing a reasonable prior typically requires work and assumptions. However, in the specific application context that we study, personalized information filtering, there is a natural way to build a prior from historical interaction data with other users. We explain and illustrate this method in Section 2.4.1 using the Yelp academic dataset (Yelp, 2012) and Section 2.4.2 using the arXiv (arXiv.org)

condensed matter dataset.

Our upper bound is an instance-specific computational upper bound on the performance of the optimal policy. It can be used to compute how far DTD-DP, DTD-UCB, or any other policy is from optimal for any given problem instance by computing the value of the heuristic with simulation, computing the upper bound, and subtracting the value from the bound. In industry, where one must allocate engineering and data science effort across projects, and one typically has a collection of concrete problems with business impact, this supports deciding whether the improvements that will be seen from continued algorithmic development are worthwhile, or whether the best existing heuristic is good enough. While our upper bound does not determine whether a proposed algorithm attains the optimal asymptotic rate, nor does it allow computing worst-case bounds over all problem instances, we argue that knowing distance from the optimal finite-time performance for specific problem instances with business impact is often more useful.

This chapter is structured as follows. In Section 2.2, we formulate the Bayesian information filtering problem. In Section 2.3, we develop a computationally tractable upper bound on the value of an optimal policy (Section 2.3.1), use this analysis to motivate development of DTD-DP (Section 2.3.2) and DTD-UCB (Section 2.3.3). In Section 2.4 we compare DTD-DP and DTD-UCB's performance against benchmarks on both real and simulated data, show a significant improvement over the best of these benchmarks, tuned UCB, and show that its performance is close to the computational upper bound across a range of problems.

2.2 Problem formulation

We consider information filtering for a single user. Items arrive to the information filtering system following a Poisson distribution with rate Γ. The n^{th} arriving item is described by a k-dimensional feature vector $X_n = (x_{1,n}, \cdots, x_{k,n})$. We assume that $x_{i,n} \geq 0$ for all i and n (If $x_{i,n}$ are bounded below, then this is without loss of generality). The vector X_n is observable to the system when the item becomes available for forwarding, and we assume the system also knows the distribution of X_n. This distribution can typically be estimated from historical data. In this chapter, we denote the density function of the feature vectors' distribution as $f(X_n)$.

Let $\theta = (\theta_1, \cdots, \theta_k)$ denote the single user's latent preference vector for the k different features. Here we model θ as having been drawn from a multivariate normal distribution with mean $\mu_0 = (\mu_{1,0}, \cdots, \mu_{k,0})$ and covariance matrix Σ_0, which represents our Bayesian prior distribution about the latent preference vector. Usually this initial belief can be obtained using the historical data from other users and we give examples of how this may be accomplished in Section 2.4.1 and Section 2.4.2. Further, we use μ_n and Σ_n to denote our Bayesian posterior distribution about the user's reward vector after the arrival of the first n items.

Upon each item's arrival, the system decides whether to forward this item to the user or not. We let $U_n \in \{0, 1\}$ represent this decision for the n^{th} item, where 1 means to forward and 0 means not to forward. If the system decides not to forward, then the item is discarded. Each time the system forwards, it pays a constant cost c and receives the item's relevance Y_n as a reward. This

relevance is modeled as the inner product between the user's unobservable vector of preferences for features θ and the item's feature vector X_n, perturbed by independent normal noise ϵ_n with variance $I(X_n)\lambda^2$, where $I(X_n)$ denotes the number of non-zero elements in X_n. The system only observes Y_n if it forwards the item. Except for the fact that some Y_n are unobserved, this statistical model is Bayesian linear regression (see Gelman et al. (2014), Chapter 14).

In many applications, $I(X_n) = k$ with probability 1, making our assumed observational variance of $I(X_n)\lambda^2$ equivalent to assuming homogeneous variance $k\lambda^2$. Even when $I(X_n)$ varies, we may modify our problem by perturbing each component of X_n by some arbitrarily small $\epsilon > 0$ to make $I(X_n) = k$ without substantially affecting the value of any particular policy.

The decision of whether or not to forward the n^{th} item can only depend on the previous information $H_{n-1} = (U_m, X_m, U_m Y_m : m \leq n-1)$ as well as our current X_n. A policy π is a sequence of functions $\pi = (\pi_1, \pi_2, \cdots)$ such that $\pi_n = (\mathbb{R}_+^k \times \{0,1\})^{n-1} \times \mathbb{R}_+^k \mapsto \{0,1\}$ and we use Π to denote the set of all such policies.

Suppose that the (random) lifetime of the user in the system is T, and let N be the total number of items that arrive to the system before T. Then our goal is to maximize:

$$\sup_{\pi \in \Pi} E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] \tag{2.1}$$

where E^{π} denotes the expected reward using policy π.

For analytic tractability, we assume that T is exponentially distributed, and let its rate parameter be $r > 0$. Then, one can show that N follows a geometric distribution with parameter $\gamma = \frac{\Gamma}{\Gamma + r}$, and the random finite horizon problem

(2.1) can be transformed to a discounted infinite horizon problem:

$$E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] = \gamma E^{\pi}\left[\sum_{n=1}^{\infty} \gamma^{n-1} U_n(Y_n - c)\right], \tag{2.2}$$

where $\gamma = \frac{\Gamma}{\Gamma + r}$. The proof is the same as Lemma 1 in Zhao & Frazier (2014) and we omit the proof here.

2.3 Main Results

The problem described in section 2.2 is a partially observable Markov decision process, and can, in theory, be solved using stochastic dynamic programming, see Lovejoy (1991) and Monahan (1982). However, the state space of this dynamic program on the belief state is in high dimension (k dimensions are required to represent the posterior mean, and $O(k^2)$ dimensions are required for the posterior covariance matrix), which makes solving it computationally intractable.

Instead, we provide in this section a computational upper bound of this problem (in Section 2.3.1) and develop two implementable policies DTD-DP and DTD-UCB based on this upper bound in Section 2.3.2 and Section 2.3.3. When DTD-DP and DTD-UCB, or any other implementable policy, gives us a result close to the upper bound, then we are reassured that this policy is nearly optimal.

In practice, DTD-DP and DTD-UCB tend to perform best when feature vectors are approximately aligned with a basis. This may tend to occur most frequently in high dimensional problems, where vectors tend to be orthogonal.

2.3.1 Upper bound

In this section, we provide a computational upper bound on the value of the solution to (2.1). This upper bound is based on the idea of dividing (2.1) into k different "single-feature" subproblems, then performing an information relaxation (similar in spirit to Brown et al. (2010)) in which we give the policy assigned to each single-feature subproblem additional information, which allows us to compute their value efficiently.

Define $Y_{i,n} = \theta_i + \epsilon_n^i$. Here $\epsilon_n^i \sim N(0, \frac{\lambda^2}{x_{i,n}^2})$ if $x_{i,n} > 0$ and $\epsilon_n^i = 0$ if $x_{i,n} = 0$ for $i = 1, \cdots, k$, independently distributed across i and n. We may think of $Y_{i,n}$ as the reward that we would have seen if X_n were equal to e_i, where e_i is a unit vector with the i_{th} element 1 and other elements 0. Later, we will use that $Y_n = \sum_{i=1}^k x_{i,n}\theta_i + \epsilon_n = \sum_{i=1}^k x_{i,n}(\theta_i + \epsilon_n^i) = \sum_{i=1}^k x_{i,n}Y_{i,n}$.

We will generalize the original problem (2.1) by introducing notation that allows for separate forwarding decisions to be made for each feature. Define $U_{j,n}$ to be decision made for the j^{th} feature of the n^{th} item. The original problem (2.1) can be recovered if we require that $U_{j,n}$ is identical across j for each n.

For each feature j, we now introduce a new set of policies Π_j, which will govern the forwarding decisions $U_{j,n}$ for feature j, and under which these decisions can depend upon information not available in the original problem: they may depend on $\theta \cdot e_i$ for $\forall i \neq j$. Formally, the decision of whether or not to forward the j^{th} feature of the n^{th} item depends on the history $H_{n-1}^j = (U_{j,m}, X_{j,m}, U_{j,m}Y_{j,m} : m \leq n-1)$, our current $X_{j,n}$, and $\theta_{-j} = (\theta_1, \cdots, \theta_{j-1}, \theta_{j+1}, \cdots, \theta_k)$.

Using these definitions, we may now state the computational upper bound. It bounds the value of the optimal policy for our original problem of interest

(2.1), on the left-hand side, by the sum of a collection of values of single-feature problems, each of which have been given additional information. Efficient computation of this right-hand side is discussed below, and summarized in Algorithm 1.

Theorem 1. For X_n that are bounded over all n, we have

$$\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] \leq \sum_{j=1}^{k} \sup_{\pi''\in\Pi_j} E^{\pi''}\left[\sum_{n=1}^{N} U_{j,n}(x_{j,n}Y_{j,n} - \frac{x_{j,n}c}{||X_n||})\right],$$

where $||X_n||$ is the L_1 norm. When $\sum_{i=1}^{k} x_{i,n} = 1$, then this theorem becomes:

$$\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] \leq \sum_{j=1}^{k} \sup_{\pi''\in\Pi_j} E^{\pi''}\left[\sum_{n=1}^{N} U_{j,n}(x_{j,n}Y_{j,n} - x_{j,n}c)\right].$$

Proof. Since $||X_n|| = x_{1,n} + \cdots + x_{k,n}$, we know

$$\begin{aligned}
&\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] \\
&= \sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N} U_n(x_{1,n}Y_{1,n} + \cdots + x_{k,n}Y_{k,n} - c)\right] \\
&= \sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N}\sum_{j=1}^{k} U_n(x_{j,n}Y_{j,n} - x_{j,n}\frac{c}{||X_n||})\right]. \qquad (2.3)
\end{aligned}$$

Now we introduce two new policy sets Π_0' and Π', which allow different features can make their own decisions $U_{j,n}$ for the n^{th} item. Further, Π_0' has an additional restriction that $U_{1,n} = \cdots = U_{j,n}$. Based on the definition, we have

$$\begin{aligned}
(2.3) &= \sup_{\pi'\in\Pi_0'} E^{\pi'}\left[\sum_{n=1}^{N}\sum_{j=1}^{k} U_{j,n}(x_{j,n}Y_{j,n} - x_{j,n}\frac{c}{||X_n||})\right] \\
&\leq \sup_{\pi'\in\Pi'} E^{\pi'}\left[\sum_{n=1}^{N}\sum_{j=1}^{k} U_{j,n}(x_{j,n}Y_{j,n} - x_{j,n}\frac{c}{||X_n||})\right]. \qquad (2.4)
\end{aligned}$$

Since the supremum of a summation is less or equal to the summation of a supremum, we have

$$(2.4) \leq \sum_{j=1}^{k} \sup_{\pi' \in \Pi'} E^{\pi'} \left[\sum_{n=1}^{N} U_{j,n}(x_{j,n} Y_{j,n} - x_{j,n} \frac{c}{||X_n||}) \right]. \tag{2.5}$$

Then based on the definition of our policy set Π_j, for $j = 1, 2, \cdots, k$, we know

$$(2.5) \leq \sum_{j=1}^{k} \sup_{\pi'' \in \Pi_j} E^{\pi''} \left[\sum_{n=1}^{N} U_{j,n}(x_{j,n} Y_{j,n} - x_{j,n} \frac{c}{||X_n||}) \right],$$

which concludes the proof of the theorem.

□

We emphasize that this computational upper bound holds true in general, even when the different components of X_n are correlated. Numerical experiments in Section 2.4 suggest that the optimality gap between this upper bound and the best heuristic policy is typically small.

For simplicity, in this chapter we focus on the special case where $\sum_{i=1}^{k} x_{i,n} = 1$. We now discuss computation of the upper bound in Theorem 1. To compute this quantity, we must solve these k subproblems:

$$\sup_{\pi \in \Pi_j} E^{\pi} \left[\sum_{n=1}^{N} U_{j,n}(x_{j,n} Y_{j,n} - x_{j,n} c) \right], j = 1, 2, \cdots, k, \tag{2.6}$$

where $Y_{j,n}|\theta_j \sim N(\theta_j, \frac{\lambda^2}{x_{j,n}^2})$ and $\theta_j \sim N(\mu_{j,n}, \sigma_{j,n}^2)$. Here $\theta_j \sim N(\mu_{j,n}, \sigma_{j,n}^2)$ represents our belief of θ_j after the first n items.

Therefore for each subproblem, after the arrival of the n^{th} item, we can update our parameters as the following:

$$\mu_{j,n} = \begin{cases} \frac{\lambda^2 \beta_{j,n-1} \mu_{j,n-1} + Y_{j,n-1} x_{j,n-1}^2}{\lambda^2 \beta_{j,n-1} + x_{j,n-1}^2} & \text{if } U_{j,n-1} = 1; \\ \mu_{j,n-1} & \text{if } U_{j,n-1} = 0. \end{cases}$$

The precision of our beliefs (which is the inverse of the prior/posterior variance with initial value $\beta_{j,0} = \frac{1}{\sigma_{j,0}^2}$) is updated as follows:

$$\beta_{j,n} = \begin{cases} \beta_{j,n-1} + \frac{x_{j,n-1}^2}{\lambda^2} & \text{if } U_{j,n-1} = 1; \\ \beta_{j,n-1} & \text{if } U_{j,n-1} = 0. \end{cases}$$

The j^{th} single-feature subproblem can be solved using dynamic programming with a three-dimensional state space $(\mu_{j,n}, \sigma_{j,n}, x_{j,n})$, where $\mu_{j,n}$ and $\sigma_{j,n}$ are the mean and variance of our current belief about θ_j and $x_{j,n}$ is the current item's j^{th} feature. Initially, $\mu_{j,0}$ and $\sigma_{j,0}$ are given by the conditional distribution of θ_j given θ_{-j} and the prior distribution $\theta \sim N(\mu, \Sigma)$. Upon each item's arrival, we move to another state based on the updating formula described above. Define $Q_j(\mu, \sigma, x, 0)$ and $Q_j(\mu, \sigma, x, 1)$ be the total reward to go if you decided to discard the item and forward the item respectively,

$$Q_j(\mu, \sigma, x, U) = \sup_{\pi'' \in \Pi_j} E^{\pi''}[\sum_{n=1}^{\infty} \gamma^{n-1} U_{j,x}(x_{j,n} Y_{j,n} - x_{j,n} c) \\ |\theta_j \sim N(\mu, \sigma^2), x_{j,1} = x, U_{j,1} = U].$$

Then the Bellman equation for this problem is:

$$V_j(\mu, \sigma, x) = \max_{U=0,1} Q_j(\mu, \sigma, x, U). \tag{2.7}$$

This calculation is summarized as Algorithm 1.

We may improve our upper bound by taking its minimum with a hindsight upper bound, derived in the following way. We first consider a larger class of policies that may additionally base their decisions on full knowledge of θ. An optimal policy among this larger class of policies forwards the n^{th} item to the

Algorithm 1 Calculation of the j^{th} subproblem

Solve the dynamic program using backward induction (discretizing and truncating), with state space $(\mu_{j,n}, \sigma_{j,n}, x_{j,n}) \in \mathbb{R} \times \mathbb{R}^+ \times [0,1]$, infinite horizon and value function $V_j(\mu, \sigma, x)$.
for $i = 1; i < M; i++$ **do**
  Generate $\theta \sim N(\mu, \Sigma)$;
  Calculate the conditional distribution of $\theta_j \sim N(\mu_{j,0}, \sigma_{j,0})$, given $\theta \sim N(\mu, \Sigma)$ and θ_{-j}.
  Generate $x_{j,0}$ from the distribution of X_n.
  Find the optimal value of state $(\mu_{j,0}, \sigma_{j,0}, x_{j,0})$ and denote it as V_i.
end for
Calculate $\bar{V} = \frac{1}{M}\sum_{i=1}^{M} V_i$ and use (2.2) to get the optimal value for the j^{th} subproblem, where M is the number of simulation.

user only if $\theta \cdot X_n > c$, and the expected total reward of this optimal policy is

$$E\left[\sum_{n=1}^{N}(\theta \cdot X_n - c)^+\right] = \frac{\gamma}{1-\gamma}E\left[(\theta \cdot X_1 - c)^+\right]. \tag{2.8}$$

Since (2.8) is the supremum of the same objective as (2.2), but over a larger set of policies, it forms an upper bound. This style of analysis was also applied in Chick & Frazier (2012). In Section 2.4, we use the minimum of the computational upper bound in Theorem 1 and the hindsight upper bound (2.8) as our theoretical upper bound.

2.3.2 The DTD-DP policy

The analysis in Section 2.3.1 provides a way to bound the performance of any policy, and is derived by decomposing the original multi-feature problem into many single-feature subproblems. In this section, we build on this same idea to develop an implementable policy, called DTD-DP, and in Section 2.3.3 we build on this idea further to create a second implementable policy, called DTD-UCB.

In DTD-DP, as each item arrives, we consider the decomposition from

Section 2.3.1 taking the incoming feature vector X_n and choosing a basis for which X_n is a unit vector in the basis. This basis may change with each n.

We then consider the decomposed problem studied in Section 2.3.1, in which we may make separate forwarding decisions for each direction in the basis, and compute the value of exploration corresponding to X_n in this decomposed problem.

To compute this value of exploration, we first compute the distribution of the magnitude x of the projection of future feature vector X along direction X_n, $x = \frac{X_n \cdot X}{X_n \cdot X_n}$, by using the distribution of future feature vectors $f(X)$. Denote this distribution by $G(x|X_n)$. We then solve the corresponding single-feature subproblem using (2.7) as described in Section 2.3.1.

From this solution, we derive Q factors, $Q(\mu_{1,0}, \sigma_{1,0}, x_0, 0)$ and $Q(\mu_{1,0}, \sigma_{1,0}, x_0, 1)$ corresponding to the value of discarding and forwarding the current item in the single feature subproblem, given that the current feature vector has magnitude $x_0 = 1$ and given that our current prior mean and variance for the subproblem are

$$\mu_{1,0} = X_n \cdot \mu_n, \sigma^2_{j,0} = X_n \Sigma_n X_n^T.$$

We then define the "exploration benefit" $E(\mu_{1,0}, \sigma_{1,0})$ from forwarding the current item as the overall benefit of forwarding, minus the myopic benefit of forwarding $\mu_{1,0} - c$ and the benefit of discarding:

$$\begin{aligned} E(\mu_{1,0}, \sigma_{1,0}) =& Q(\mu_{1,0}, \sigma_{1,0}, 1, 1) - \\ & Q(\mu_{1,0}, \sigma_{1,0}, 1, 0) - \mu_{1,0} + c. \end{aligned}$$

In DTD-DP, we add a scalar tuning parameter α, mirroring the tuning

parameter used in UCB, to scale up or down the exploration benefit. The default value for α is $\alpha = 1$. Then, returning to the original multi-dimensional problem, we consider the net benefit of forwarding to be the myopic benefit $X_n \cdot \mu_n - c$ plus the exploration benefit $\alpha E(\mu_{1,0}, \sigma_{1,0})$, and forward when this is strictly positive. This is summarized in Algorithm 2.

Algorithm 2 The DTD-DP algorithm

for $n = 1, 2, \cdots$ **do**
  Denote $\mu_{1,0} = X_n \cdot \mu_n$ and $\sigma^2_{1,0} = X_n \Sigma_n X_n^T$;
  Calculate $Q(\mu_{1,0}, \sigma_{1,0}, 1, U)$ for $U = 0, 1$ given that $x \sim G(x|X_n)$;
  Denote $E(\mu_{1,0}, \sigma_{1,0}) = Q(\mu_{1,0}, \sigma_{1,0}, 1, 1) - Q(\mu_{1,0}, \sigma_{1,0}, 1, 0) - \mu_{1,0} + c$;
  if $\mu_{1,0} + \alpha \cdot E(\mu_{1,0}, \sigma_{1,0}) > c$ **then**
    Forward the item
  else
    Discard the item
  end if
end for

2.3.3 The DTD-UCB algorithm

In this section, we develop a second heuristic, DTD-Upper-Confidence-Bound (DTD-UCB), which builds on the ideas underlying DTD-DP.

In DTD-DP, we considered a single-feature subproblem in which the magnitude x of the projection of future feature vectors is given by $G(x|X_n)$ and in which the prior mean and prior variance were given by $X_n \cdot \mu_n$ and $X_n \Sigma_n X_n^T$ respectively. We then quantified the value of exploration by solving the single-feature subproblem using stochastic dynamic programming. In this single-feature subproblem, we observe that when future feature vectors are more closely aligned with X_n, so that samples from $G(x|X_n)$ are large, we are more willing to explore.

In our second heuristic DTD-UCB, we take a similar approach, but quantify the value of exploration using an approach adopted from the literature on upper confidence bound policies, which quantifies the value of exploration in terms of some scalar multiple α of the standard deviation of the value of an action, obtained from calculating an upper confidence bound and subtracting the center of the confidence region. In DTD-UCB, we quantify the value of information similarly, but add an additional scaling factor to include the fact that those X_n whose $G(x|X_n)$ have larger moments should induce more exploration.

To accomplish this, we let $M(X_n)$ be the mean of the distribution $G(x|X_n)$. This "mean of the projection" is

$$M(X_n) = \int_X \frac{X_n \cdot X}{X_n \cdot X_n} f(X) dX.$$

We summarize the DTD-UCB algorithm in Algorithm 3.

Algorithm 3 The DTD-UCB algorithm

for $n = 1, 2, \cdots$ **do**
  if $X_n \cdot \mu_n + \alpha \cdot M(X_n) \cdot \sqrt{X_n \Sigma_n X_n} > c$ **then**
    Forward the item
  else
    Discard the item
  end if
end for

2.4 Numerical Experiments

In this section, we compare DTD-DP and DTD-UCB with three different benchmark algorithms and the computational upper bound from Section 2.3.1 using both real and simulated data. The benchmark algorithms are:

- Pure Exploitation: Forward the item if $X_n \cdot \mu_n \geq c$.
- Upper Confidence Bound (UCB): Forward the item if $X_n \cdot \mu_n + \alpha \sqrt{X_n \Sigma_n X_n^T} \geq c$.
- Linear Thompson Sampling (LTS): For item X_n, generate $\theta \sim N(\mu_n, \Sigma_n)$. Forward the item if $\theta \cdot X_n > c$.

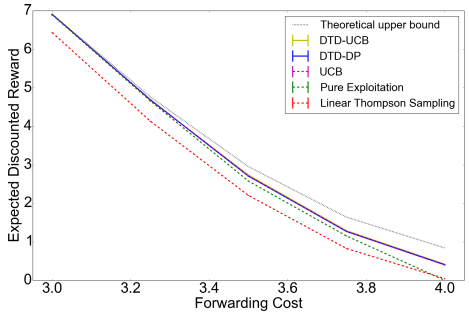
For DTD-DP, DTD-UCB and UCB, there is a tuning parameter α. In our simulation experiments we run these policies with 10 different values of α ranging from 0.1 to 10 on a log scale, and display the one with the best performance (which requires simulating performance for different values of α in a Monte Carlo simulation as a pre-processing step) in each instance.

We evaluate our upper bound and proposed policy on real and simulated data, and find our upper bound is tight enough to be useful (the best policy evaluated is often within 60% of the upper bound and never below 30% of the upper bound).

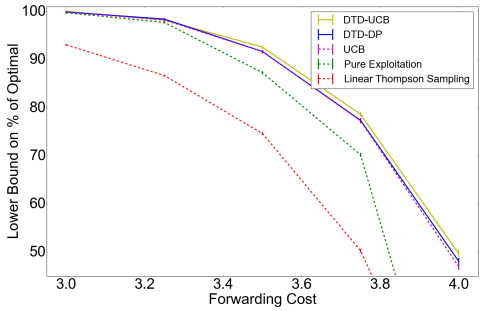
2.4.1 Yelp academic data

In this section, we compare DTD-DP and DTD-UCB against benchmarks using the Yelp academic dataset (Yelp, 2012).

Our items are businesses, and are described as belonging to one or more of the following six categories: Restaurants, Shopping, Food, Beauty and Spas, Health and Medical and Nightlife. The j^{th} business object is then described by a 6-dimensional feature vector $X_j = (x_{1,j}, x_{2,j}, \cdots, x_{6,j})$ with the i^{th} element $x_{i,j} = 1$

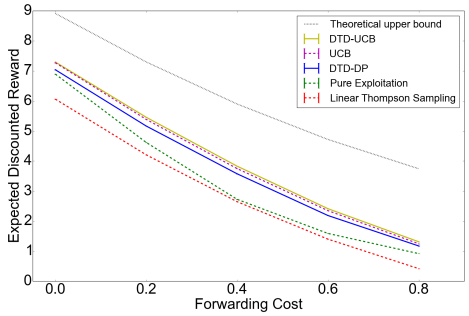


(a) Comparison of Different Policies Using Yelp Academic Data

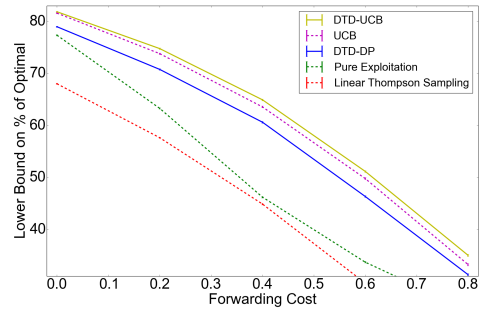


(b) Optimality Gap of Different Policies Using Yelp Academic Data

Figure 2.1: The performance of DTD-DP, DTD-UCB and three benchmark algorithms relative to the computational upper bound. This plot compares performance on the Yelp academic dataset (Section 2.4.1), and shows that DTD-UCB outperforms all other heuristic policies. DTD-DP performs comparably (and nearly identical to) UCB, and outperforms pure exploitation and LTS. DTD-UCB performs close to the computational upper bound, showing their performance is close to optimal.



(a) Comparison of Different Policies Using arXiv.org dataset

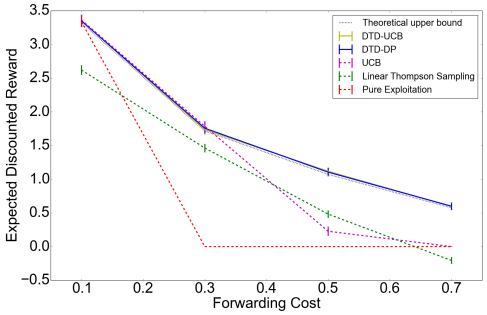


(b) Optimality Gap of Different Policies Using arXiv.org dataset

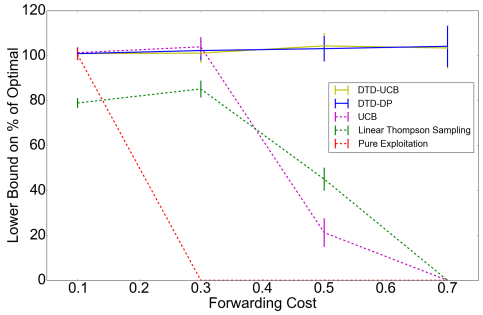
Figure 2.2: The performance of DTD-DP, DTD-UCB and three benchmark algorithms relative to the computational instance-specific upper bound. This plot compares performance on the 2014 arXiv.org Condensed Matter dataset (Section 2.4.2), and shows that DTD-UCB outperforms all other heuristic policies.

if the business belongs to category i, and $x_{i,j} = 0$ otherwise. Then we normalize X_j such that its L1 norm is 1.

We calculate the prior distribution over new customers' preferences using



(a) Comparison of Different Policies Using Simulated Data



(b) Optimality Gap of Different Policies Using Simulated Data

Figure 2.3: The performance of DTD-DP, DTD-UCB and three benchmark algorithms relative to the computational instance-specific upper bound using simulated data. This plot compares performance on simulated data (Section 2.4.3), and shows that DTD-DP and DTD-UCB outperform all the other algorithms and coincides with the theoretical upper bound, showing it is indistinguishable from optimal in this case.

historical users' reviews. For each historical user, we use linear regression to regress his reviews' ratings on the feature vectors of the business objects that he reviewed. We use the estimated linear regression coefficients as his/her true user preference vector. Then we calculate the empirical distribution for all historical users, and set the prior on new users' preference vectors to be multivariate normal with mean vector and covariance matrix equal to the sample mean and sample covariance of the historical users.

In Figure 2.1a, evaluation is done by taking a collection of real historical users, and for each estimating his true preference vector θ using linear regression on historical data. Evaluation is then performed for each algorithm and user by simulating feedback from the user's held out θ on items forwarded by the algorithm, and an algorithm's average performance is calculated by averaging across users. We must simulate user feedback given θ because we do not have historical relevance feedback from all users for all items, and

algorithms may present items that have not been rated. We plot the 95% confidence interval of cumulative reward over 100 items forwarded to the user with discount factor $\lambda = 0.9$.

In Figure 2.1b, we calculate the optimality gap between each heuristic algorithm and our computational upper bound. A smaller gap suggests the corresponding policy performs better in this problem instance.

The plot in Figure 2.1 summarizes the results. In this problem instance, DTD-UCB outperforms DTD-DP, UCB, pure exploitation and LTS, with DTD-DP and UCB performing almost identically. Moreover, the optimality gap is relatively small, which shows that DTD-UCB performs close to optimal.

2.4.2 arXiv.org Condensed Matter Dataset

In this section, we compare DTD-DP and DTD-UCB with benchmarks using readership data from articles submitted in 2014 to the arXiv condensed matter category. We represent each paper submitted in 2014 by a 10 dimensional vector using Latent Dirichlet allocation (LDA) (Blei et al., 2003). For each user, the rating for a paper is 1 if he/she clicks and otherwise the rating is 0. We then calculate the user's preference vector by linear regression. Similar to Section 2.4.1, we use the sample mean and sample variance of users' preference vectors as our prior distribution parameters.

In our simulation, we use true users' preference vectors calculated using linear regression, as we did in Section 2.4.1. For each user, we randomly pick 100 papers and make the forwarding decisions using different policies. We evaluate

the cumulative reward for these 100 papers with discount factor $\lambda = 0.9$.

The result is summarized in Figure 2.2. The best of our heuristic policies in this example, DTD-UCB, outperforms all other heuristic policies. In this specific example, DTD-DP does not perform as well as UCB but it outperforms pure exploitation and LTS.

2.4.3 Simulated Data

In this section, we compare the performance of DTD-DP and DTD-UCB with three benchmark algorithms, as well as our computational upper bound on simulated data. This simulated data is chosen to give insight into situations where UCB can underperform, and where the structure of a policy like DTD-DP and DTD-UCB are needed to provide near-optimal performance. We emphasize that it is chosen to provide insight, and not to show performance on a typical real problem instance — we refer this comparison to Section 2.4.1 and Section 2.4.2.

Each item is described by a 100-dimensional feature vector X_n with the following distribution: $P(X_n = e_1) = \frac{100}{199}$, $P(X_n = e_i) = \frac{1}{199}$ for $i = 2, \cdots, 100$. Here, e_x is the unit vector in the xth dimension. The initial belief on the user's preference for each feature is $N(0.3, 1.0)$ with independence across features. We set $\gamma = 0.9$ and $\lambda = 0.1$. In estimating the infinite-horizon discounted sum (2.2), we truncate after $n = 100$.

The results, summarized in Figure 2.3, show that DTD-DP and DTD-UCB outperform UCB, pure exploitation and LTS. In most cases, UCB performs very

well with a properly chosen α. Moreover, DTD-DP and DTD-UCB outperform UCB for several values of the forwarding cost, and nearly coincides with the theoretical upper bound for all values of the forwarding cost, which shows that it is indistinguishable from optimal in this problem instance.

LTS does not perform well in this example because it performs poorly at the initial stages and the (discounted) reward in the later stages cannot make up for the loss at the early stages. As Russo & Van Roy (2014) and Russo & Van Roy (2014) pointed out, LTS generally underperforms tuned UCB.

UCB underperforms DTD-DP and DTD-UCB in this example because it cannot account for the frequency with which a feature appears, and thus cannot adjust its level of exploration (encoded as the choice of α) to explore more those features that tend to reoccur frequently, and explore less those features that are unlikely to appear again. In contrast, both DTD-DP and DTD-UCB can adjust its level of exploration, and will explore more those features that will reoccur.

2.5 Conclusion

We studied the Bayesian linear information filtering problem, providing an instance-specific computational upper bound and a pair of new *Decompose-Then-Decide* heuristic policies, DTD-DP and DTD-UCB. Numerical experiments show that the best of these two policies is typically close to the computational upper bound and outperforms several benchmarks on real and simulated data.

CHAPTER 3

DUELING BANDITS WITH WEAK REGRET

3.1 Introduction

In this Chapter, we study ordinal feedback with application in personalized content recommendation systems. We offer pairs of items to a user and record implicit feedback on which offered item is preferred, seeking to learn the user's preferences over items quickly, while also ensuring that the fraction of time we fail to offer a high-quality item is small. Implicit pairwise comparisons avoid the inaccuracy of user ratings (Joachims et al., 2007) and the difficulty of engaging users in providing explicit feedback.

We study a model for this setting called the dueling bandit problem (Yue & Joachims, 2009). The items we may offer to the user are called "arms", and we learn about these arms through a sequence of "duels". In each duel, we "pull" two arms and receive noisy feedback from the user telling us which arm is preferred. When an arm is preferred within a duel, we say that the arm has "won the duel".

We study this problem in the Condorcet winner setting, in which we assume the existence of an arm (the Condorcet winner) that wins with probability at least $\frac{1}{2}$ when paired with any of the other arms. In these settings, we consider two notions of regret: "weak regret", in which we avoid regret by selecting the Condorcet winner as either arm in the duel; and "strong-regret", in which we can only avoid regret by setting both arms in the duel to the Condorcet winner.

Weak regret was proposed by Yue et al. (2012) and arises in content

recommendation when arms correspond to items, and the user incurs no regret whenever his most preferred item is made available. Examples include in-app restaurant recommendations provided by food delivery services like Grubhub and UberEATS, in which implicit feedback may be inferred from selections, and the user only incurs regret if her most preferred restaurant is not recommended. Examples also include recommendation of online broadcasters on platforms such as Twitch, in which implicit feedback may again be inferred from selections, and the user is fully satisfied as long as her favored broadcaster is listed. Despite its applicability, Yue et al. (2012) is the only paper of which we are aware that studies weak regret, and it does not provide algorithms specifically designed for this setting.

Strong regret has been more widely studied, as discussed below, and has application to choosing ranking algorithms for search (Hofmann et al., 2013). To perform a duel, query results from two rankers are interleaved (Radlinski et al., 2008), and the ranking algorithm that provided the first result chosen by the user is declared the winner of the duel. Strong regret is appropriate in this setting because the user's experience is enhanced by pulling the best arm twice, so that all of that ranker's results are shown.

Our contribution is a new algorithm, *Winner Stays* (WS), with variants designed for the weak (WS-W) and strong regret (WS-S) settings. We prove that WS-W has expected cumulative weak regret that is constant in time, with dependence on the number of arms N given by $O(N^2)$. If the arms have a total order, we show a tighter bound of $O(N \log N)$. We then prove that WS-S has expected cumulative strong regret that is $O(N^2 + N \log(T))$, and prove that a tighter bound of $O(N \log(N) + N \log(T))$ holds if arms have a total order. These

regret bounds are optimal in T, and for weak regret are strictly better than those for any previously proposed algorithm, although at the same time both strong and weak regret bounds are sensitive to the minimum gap in winning probability between arms. We demonstrate through numerical experiments on simulated and real data that WS-W and WS-S significantly outperform existing algorithms on strong and weak regret.

The chapter is structured as follows. Section 3.2 reviews related work. Section 3.3 formulates our problem. Section 3.4 introduces the *Winner Stays* (WS) algorithm: Section 3.4.1 defines WS-W for the weak regret setting; Section 3.4.2 proves that WS-W has cumulative expected regret that is constant in time; Section 3.4.3 defines WS-S for the strong regret setting and bounds its regret. Section 3.4.4 disusses a simple extension of our theoretical results to the utility-based bandit setting, which is used in our numerical experiments. Section 3.5 compares WS with three benchmark algorithms using both simulated and real datasets, finding that WS outperforms these benchmarks on the problems considered.

3.2 Related Work

Most work on dueling bandits focuses on strong regret. Yue et al. (2012) shows that the worst-case expected cumulative strong regret up to time T for any algorithm is $\Omega(N \log(T))$. Algorithms have been proposed that reach this lower bound under the Condorcet winner assumption in the finite-horizon setting: Interleaved Filter (IF) (Yue et al., 2012) and Beat the Mean (BTM) (Yue & Joachims, 2011). Relative Upper Confidence Bound (RUCB) (Zoghi

et al., 2014) also reaches this lower bound in the horizonless setting. Relative Minimum Empirical Divergence (RMED) (Komiyama et al., 2015) is the first algorithm to have a regret bound that matches this lower bound. Zoghi et al. (2015) proposed two algorithms, Copeland Confidence Bound (CCB) and Scalable Copeland Bandits (SCB), which achieve an optimal regret bound without assuming existence of a Condorcet winner.

While weak regret was proposed in Yue et al. (2012), it has not been widely studied to our knowledge, and despite its applicability we are unaware of papers that provide algorithms designed for it specifically. While one can apply algorithms designed for the strong regret setting to weak regret, and use the fact that strong dominates weak regret to obtain weak regret bounds of $O(N \log(T))$, these are looser than the constant-in-T bounds that we show.

Active learning using pairwise comparisons is also closely related to our work. Jamieson & Nowak (2011) considers an active learning problem that is similar to our problem in that the primary goal is to sort arms based on the user's preferences, using adaptive pairwise comparisons. It proposes a novel algorithm, the Query Selection Algorithm (QSA), that uses an expected number of operations of $d \log(N)$ to sort N arms, where d is the dimension of the space in which the arms are embedded, rather than $N \log(N)$. Busa-Fekete et al. (2013) and Busa-Fekete et al. (2014) consider top-k element selection using adaptive pairwise comparisons. They propose a generalized racing algorithm focusing on minimizing sample complexity. Pallone et al. (2017) studies adaptive preference learning across arms using pairwise preferences. They show that a greedy algorithm is Bayes-optimal for an entropy objective. While similar in that they use pairwise comparisons, these algorithms are different in focus from

the current work because they do not consider cumulative regret.

3.3 Problem Formulation

We consider N items (arms). At each time $t = 1, 2, \ldots$, the system chooses two items and shows them to the user, i.e., the system performs a duel between two arms. The user then provides binary feedback indicating her preferred item, determining which arm wins the duel. This binary feedback is random, and is conditionally independent of all past interactions given the pair of arms shown. We let $p_{i,j}$ denote the probability that the user gives feedback indicating a preference for arm i, when shown arms i and j. If the user prefers arm i over arm j, we assume $p_{i,j} > 0.5$. We also assume symmetry: $p_{i,j} = 1 - p_{j,i}$.

We assume arm 1 is a Condorcet winner, i.e., that $p_{1,i} > 0.5$ for $i = 2, \cdots, N$. In some results, we also consider the setting in which arms have a total order, by which we mean that the arms are ordered so that $p_{i,j} > 0.5$ for all $i < j$. The total order assumption implies transitivity.

We let $p = \min_{p_{i,j}>0.5} p_{i,j} > 0.5$ be a lower bound on the probability that the user will choose her favourite arm.

We consider both weak and strong regret in its binary form. The single-period *weak regret* incurred at this time is $r(t) = 1$ if we do not pull the best arm and $r(t) = 0$ otherwise. The single-period *strong regret* is $r(t) = 1$ if we do not pull the best arm twice and $r(t) = 0$ otherwise. We also consider utility-based extensions of weak and strong regret in Section 3.4.4.

We use the same notation $r(t)$ to denote strong and weak regret, and rely

on context to distinguish the two cases. In both cases, we define the cumulative regret up to time T to be $R(T) = \sum_{t=1}^{T} r(t)$. We measure the quality of an algorithm by its expected cumulative regret.

3.4 Winner Stays

We now propose an algorithm, called *Winner Stays* (WS), with two variants: WS-W designed for weak regret; and WS-S for strong regret. Section 3.4.1 introduces WS-W and illustrates its dynamics. Section 3.4.2 proves the expected cumulative weak regret of WS-W is $O(N^2)$ under the Condorcet winner setting, and $O(N \log(N))$ under the total order setting. Section 3.4.3 introduces WS-S and proves that its expected cumulative strong regret is $O(N^2 + N \log(T))$ under the Condorcet winner setting, and $O(N \log(T) + N \log(N))$ under the total order setting, both of which have optimal dependence on T. Section 3.4.4 extends our theoretical results to utility-based bandits.

3.4.1 Winner Stays with Weak Regret (WS-W)

We now present WS-W, first defining some notation. Let $q_{i,j}(t)$ be the number of times that arm i has defeated arm j in a duel, up to and including time t. Then, define $C(t, i) = \sum_{j \neq i} q_{i,j}(t) - q_{j,i}(t)$. $C(t, i)$ is the difference between the number of duels won and lost by arm i, up to time t. With this notation, we define WS-W in Algorithm 4.

WS-W's pulls can be organized into *iterations*, each of which consists of a sequence of pulls of the same pair of arms, and *rounds*, each of which consists

Algorithm 4 WS-W

Input: arms $1, \cdots, N$.
for $t = 1, 2, \cdots$ **do**
Step 1: Pick $i_t = \arg\max_i C(t-1, i)$, breaking ties as follows:

- If $t > 1$ and $i_{t-1} \in \arg\max_i C(t-1, i)$, set $i_t = i_{t-1}$.
- Else if $t > 1$ and $j_{t-1} \in \arg\max_i C(t-1, i)$, set $i_t = j_{t-1}$.
- Else choose i_t uniformly at random from $\arg\max_i C(t-1, i)$.

Step 2: Pick $j_t = \arg\max_{j \neq i_t} C(t-1, j)$, breaking ties as follows:

- If $t > 1$ and $i_{t-1} \in \arg\max_{i \neq i_t} C(t-1, i) \setminus \{i_t\}$, set $j_t = i_{t-1}$.
- Else if $t > 1$ and $j_{t-1} \in \arg\max_{i \neq i_t} C(t-1, i) \setminus \{i_t\}$, set $j_t = j_{t-1}$.
- Else choose j_t uniformly at random from $\arg\max_j C(t-1, j) \setminus \{i_t\}$.

Step 3: Pull arms i_t and j_t;
Step 4: Observe noisy binary feedback and update $C(t, i_t)$ and $C(t, j_t)$;
end for

of a sequence of iterations in which arms that lose an iteration are not visited again until the next round. We first describe iterations and rounds informally with an example and in Figure 3.1 before presenting our formal analysis.

Example: At time $t = 1$, $C(0, i) = 0$ for all i, and WS-W pulls two randomly chosen arms. Suppose it pulls arms $i_1 = 1$, $j_1 = 2$ and arm 1 wins. Then $C(1, i)$ is 1 for arm 1, -1 for arm 2, and 0 for the other arms. This first pull is an iteration of length 1, arm 1 is the winner, and arm 2 is the loser. This iteration is in the first round. We call $t_1 = 1$ the start of the first round, and $t_{1,1} = 1$ the start of the first iteration in the first round.

At time $t = 2$, $C(t-1, i)$ is largest for arm 1 so WS-W chooses $i_2 = 1$. Since $C(t-1, i)$ is -1 for arm 2 and 0 for the other arms, WS-W chooses j_2 at random from arms 3 through N (suppose $N > 2$). Suppose it chooses arm $j_2 = 3$. This pair of arms (1 and 3) is different from the pair pulled in the previous iteration

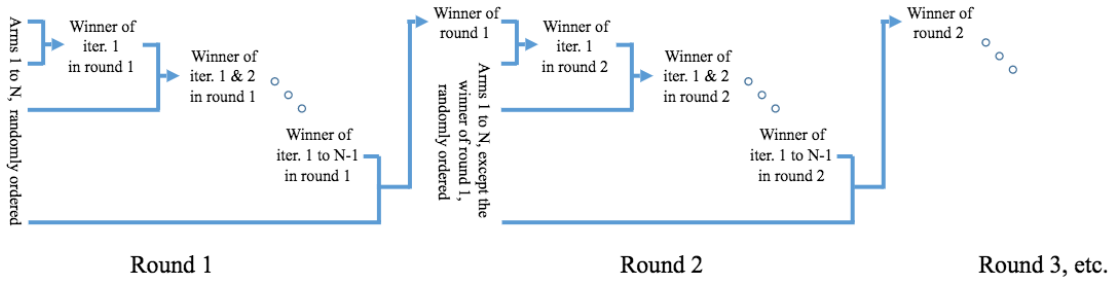


Figure 3.1: Our analysis of WS-W decomposes its behavior into a sequence of rounds. In each round, pairs of arms play each other in a sequence of iterations. The winner from an iteration passes on to play a new arm in the next iteration randomly selected from those that have not yet played in the round. At the end of a round, the round's winner is considered first in the next round.

(1 and 2), so $t_{1,2} = 2$ is the start of the second iteration (in the first round).

WS-W continues pulling arms 1 and 3 until $C(t, i)$ is -1 for one of these arms and 2 for the other. WS-W continues to pull only arms 1 and 3 until one has $C(t, i) = 2$ even though this may involve times when $C(t, i)$ is 0 for both arms 1 and 3, causing them to be tied with arms 4 and above, because we break ties to prioritize pulling previously pulled arms. The sequence of times when we pull arms 1 and 3 is the second iteration. The arm that ends the iteration with $C(t, i) = 2$ is the winner of that iteration.

WS-W continues this process, performing $N - 1$ iterations on different pairs of arms, pitting the winner of each iteration against a previously unplayed arm in the next iteration. This sequence of iterations is the first round. The winner of the final iteration in the first round, call it arm $Z(1)$, has $C(t, Z(1)) = N - 1$ and all other arms $j \neq Z(1)$ have $C(t, j) = -1$.

The second round begins on the next pull after the end of the first round, at time t_2. WS-W again performs $N - 1$ iterations, playing $Z(1)$ in the first iteration. Each iteration has a winner that passes to the next iteration.

WS-W repeats this process for an infinite number of rounds. Each round is a sequence of $N-1$ iterations, and an arm that loses an iteration is not revisited until the next round. Figure 3.1 illustrates these dynamics, and we formalize the definition of round and iteration in the next section.

3.4.2 Analysis of WS-W

In this section, we analyze the weak regret of WS-W. After presenting definitions and preliminary results, we prove WS-W has expected cumulative weak regret bounded by $O(N\log(N))$ when arms have a total order. Then, in the more general Condorcet winner setting, we prove WS-W has expected cumulative weak regret bounded by $O(N^2)$. We leave the proofs of all lemmas to the supplement.

We define t_ℓ, the *beginning of round* ℓ, and $Z(\ell-1)$, the *winner* of round ℓ, as the unique time and arm such $C(t_\ell-1, Z(\ell-1)) = (N-1)(\ell-1)$ and $C(t_\ell-1, i) = -\ell+1$ for all $i \neq Z(\ell-1)$.

We define $t_{\ell,k}$, the *beginning of iteration k in round* ℓ, as the first time we pull the k^{th} unique pair of arms in the ℓ^{th} round. We let $T_{\ell,k}$ be the number of successive pulls of this pair of arms.

We additionally define terminology to describe arms pulled in an iteration. In a duel between arms i and j with $p_{i,j} > 0.5$, arm i is called the *better arm* and arm j is called the *worse arm*. We say that an arm i is the *incumbent* in iteration k iteration and round ℓ if $C(t_{\ell,k}-1,i) > 0$. A unique such arm exists except when $\ell = k = 1$. When $\ell = k = 1$, the incumbent is the better of the two arms being played. We call the arm being played that is not the incumbent the *challenger*.

Using these definitions, we present our first pair of results toward bounding the expected cumulative weak regret of WS-W. They bound the number of pulls in an iteration.

Lemma 1. The conditional expected length of iteration k in round ℓ, given the arms being pulled, is bounded above by $\frac{N(\ell-1)+k}{2p-1}$ if the incumbent is worse than the challenger, and by $\frac{1}{2p-1}$ if the incumbent is better than the challenger.

Lemma 1 shows that iterations with a worse incumbent use more pulls. We then bound the number of iterations with a worse incumbent.

Lemma 2. Under the total order assumption, the conditional expected number of future iterations with an incumbent worse than the challenger, given history up to time $t_{\ell,k}$, is bounded above by $\frac{2p^2}{(2p-1)^3}(\log(N)+1)$ for any $k, \ell \geq 1$.

Lemma 2 implies that the incumbent is worse than the challenger in finitely many iterations with probability 1. We now bound the tail distribution of the last such round.

Lemma 3. Let L denote the smallest ℓ such that no round $\ell' > \ell$ contains an iteration in which the incumbent is worse than the challenger. Then $P(L \geq \ell) \leq \left(\frac{1-p}{p}\right)^{\ell}$.

To present our final set of preliminary lemmas, we define several indicator functions. Let $B(\ell, k)$ be 1 when the incumbent in iteration k of round ℓ is better than the challenger. Let $D(\ell)$ be 1 if arm 1 (the best arm) is the incumbent at the beginning of iteration 1 of round ℓ. Denote $\bar{B}(\ell, k) = 1 - B(\ell, k)$ and $\bar{D}(\ell) = 1 - D(\ell)$. Let $V(\ell, k)$ be 1 if $D(\ell) = 1$ and arm 1 loses in any iteration 1 through $k-1$ of round ℓ.

We may only incur weak regret during round ℓ iteration k if $\bar{D}(\ell) = 1$, or if $V(\ell, k') = 1$ for some $k' < k$. We will separately bound the regret incurred in these two different scenarios. Moreover, our bound on the number of pulls, and thus the regret incurred, in this iteration will depend on whether $B(\ell, k) = 1$ or $\bar{B}(\ell, k) = 1$. This leads us to state four inequalities in the following pair of lemmas, which we will in turn use to show Theorem 2. The first lemma applies in both the total order and Condorcet settings, while the second applies only in the total order setting. When proving Theorem 3 we replace Lemma 5 by an alternate pair of inequalities.

Lemma 4.

$$\mathbb{E}[\bar{D}(\ell)B(\ell, k)T_{\ell,k}] \leq \frac{1}{2p-1}\left(\frac{1-p}{p}\right)^{\ell-1},$$
$$\mathbb{E}[V(\ell, k)B(\ell, k)T_{\ell,k}] \leq \frac{1}{2p-1}\left(\frac{1-p}{p}\right)^{\ell}.$$

Lemma 5. Under the total order assumption:

- $\mathbb{E}\left[\sum_{k=1}^{N-1} \bar{D}(\ell)\bar{B}(\ell, k)T_{\ell,k}\right]$ is bounded above by $\left(\frac{1-p}{p}\right)^{\ell-1} \frac{2N\ell p^2}{(2p-1)^4}(\log(N) + 1)$.
- $\mathbb{E}\left[\sum_{k=1}^{N-1} V(\ell, k)\bar{B}(\ell, k)T_{\ell,k}\right]$ is bounded above by $\left(\frac{1-p}{p}\right)^{\ell} \frac{2N\ell p^2}{(2p-1)^4}(\log(N) + 1)$.

We now state our main result for the total order setting, which shows that the expected cumulative weak regret is $O\left(\frac{N\log(N)}{(2p-1)^5}\right)$.

Theorem 2. The expected cumulative weak regret of WS-W is bounded by $\left[\frac{2p^3}{(2p-1)^6}N(\log(N) + 1) + \frac{N}{(2p-1)^2}\right]$ under the total order assumption.

Proof. Iterations can be divided into two types: those in which the incumbent is better than the challenger, and those where the incumbent is worse.

We first bound expected total weak regret incurred in the first type of iteration, and then below bound that incurred in the second type. In this first bound, observe that we incur weak regret during round ℓ if $D(\ell) = 0$, or if $D(\ell) = 1$ but arm 1 loses to some other arm during this round. Under the second scenario, we do not incur any regret until arm 1 loses to another arm.

Thus, the expected weak regret incurred during iterations with a better incumbent is bounded by

$$\mathbb{E}\left[\sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1} B(\ell,k)T_{\ell,k}\bar{D}(\ell) + \sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1} B(\ell,k)T_{\ell,k}V(\ell,k)\right].$$

The first part of this summation can be bounded by the first inequality in Lemma 4 to obtain

$$\begin{aligned}&\mathbb{E}\left[\sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1} B(\ell,k)T_{\ell,k}\bar{D}(\ell)\right]\\ &\leq \sum_{\ell=1}^{\infty}\left(\frac{1-p}{p}\right)^{\ell-1}\frac{N}{2p-1} = \frac{pN}{(2p-1)^2}.\end{aligned}$$

The second part of this summation can be bounded by the second inequality in Lemma 4 to obtain

$$\begin{aligned}&\mathbb{E}\left[\sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1} B(\ell,k)T_{\ell,k}V(\ell,k)\right]\\ &\leq \sum_{\ell=1}^{\infty}\frac{N}{2p-1}\left(\frac{1-p}{p}\right)^{\ell} = \frac{N(1-p)}{(2p-1)^2}.\end{aligned}$$

Thus, the cumulative expected weak regret incurred during iterations with a better incumbent is bounded by $\frac{N}{(2p-1)^2}$.

Now we bound the expected weak regret incurred during iterations where the incumbent is worse than the challenger. This is bounded by

$$\mathbb{E}\left[\sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1} \bar{B}(\ell,k)T_{\ell,k}\bar{D}(\ell) + \sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1} \bar{B}(\ell,k)T_{\ell,k}V(\ell,k)\right].$$

The first term in the summation can be bounded by the first inequality of Lemma 5 to obtain

$$\begin{aligned}
&\mathbb{E}\left[\sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1}\bar{B}(\ell,k)T_{\ell,k}\bar{D}(\ell)\right] \\
&\leq \sum_{\ell=1}^{\infty}\frac{2N\ell p(1-p)}{(2p-1)^4}(\log(N)+1)\left(\frac{1-p}{p}\right)^{\ell-1} \\
&= \frac{2Np^4}{(2p-1)^6}(\log(N)+1).
\end{aligned}$$

The second term in the summation can be bounded by the first inequality of Lemma 5 to obtain

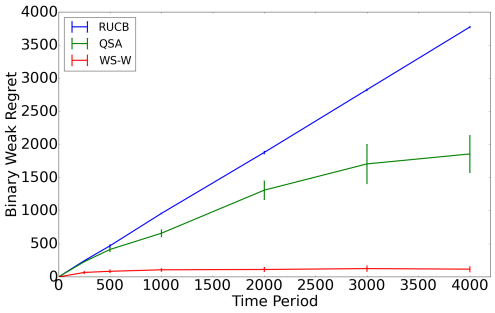
$$\begin{aligned}
&\mathbb{E}\left[\sum_{\ell=1}^{\infty}\sum_{k=1}^{N-1}\bar{B}(\ell,k)T_{\ell,k}V(\ell,k)\right] \\
&\leq \sum_{\ell=1}^{\infty}\frac{2N\ell p^2}{(2p-1)^4}(\log(N)+1)\left(\frac{1-p}{p}\right)^{\ell} \\
&= \frac{2p^3(1-p)}{(2p-1)^6}N(\log(N)+1).
\end{aligned}$$

Thus, the cumulative expected weak regret incurred during iterations with a worse incumbent is bounded by $\frac{2p^3}{(2p-1)^6}N(\log(N)+1)$.

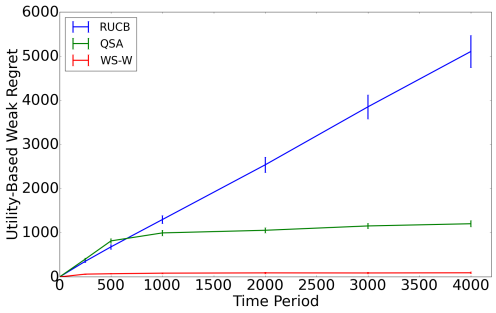
Summing these two bounds, the cumulative expected weak regret is bounded by $\left[\frac{2p^3}{(2p-1)^6}N(\log(N)+1)+\frac{N}{(2p-1)^2}\right]$. □

We prove the following result for the Condorcet winner setting in a similar manner in the supplement.

Theorem 3. The expected cumulative weak regret of WS-W is bounded by $\frac{N}{(2p-1)^2}+\frac{pN^2}{(2p-1)^3}$ under the Condorcet winner setting.



(a) Simulated Dataset



(b) Yelp Academic Dataset

Figure 3.2: Comparison of the weak regret between WS-W, RUCB and QSA using simulated data, and the Yelp academic dataset. In both experiments, WS-W outperforms RUCB and QSA, provided constant expected cumulative weak regret.

3.4.3 Winner Stays with Strong Regret (WS-S)

In this section, we define a version of WS for strong regret, WS-S, which uses WS-W as a subroutine. WS-S is defined in Algorithm 5

Algorithm 5 WS-S

Input: $\beta > 1$, arms $1, \cdots, N$.
for $\ell = 1, 2, \cdots$ **do**
  Exploration phase: Run the ℓ^{th} round of WS-W.
  Exploitation phase: Let $Z(\ell)$ be the index of the best arm at the end of the ℓ^{th} round. For the next $\lfloor \beta^{\ell} \rfloor$ time periods, pull arms $Z(\ell)$ and $Z(\ell)$ and ignore the feedback.
end for

Each round of WS-S consists of an exploration phase and an exploitation phase. The length of the exploitation phase increases exponentially with the number of phases. Changing the parameter β balances the lengths of these phases, and thus balances between exploration and exploitation. Our theoretical results below guide choosing β.

We now bound the cumulative strong regret of this algorithm under both the total order and Condorcet winner settings:

Theorem 4. If there is a total order among arms, then for $1 < \beta < \frac{p}{1-p}$, the expected cumulative strong regret for WS-S is bounded by $\left[\frac{2p^3}{(2p-1)^6}N(\log(N)+1) + \frac{N\log_\beta(T(\beta-1))}{2p-1}\right]$.

Proof. Suppose at time T, we are in round ℓ. Then $\beta + \cdots + \beta^\ell \leq T$. Solving for ℓ, we obtain $\ell \leq \log_\beta(T(\beta-1))$.

We bound the expected strong regret up to time T. The expected regret can be divided in two parts: the regret occuring during the exploration phase; and the regret occuring during the exploitation phase.

First we focus on regret incurred during exploration. We never pull the same arm twice during this phase, and so regret is incurred in each time period. To bound regret incurred during exploration, we bound the length of time spent in this phase.

The length of time spent in exploration up to the end of round ℓ with a better incumbent is bounded by $\frac{(N-1)\ell}{2p-1}$. The length of time spent with a worse incumbent, based on the proof of Theorem 2, is bounded by $\frac{2p^3}{(2p-1)^6}N(\log(N)+1)$.

Now we focus on regret incurred during exploitation. The probability we have identified the wrong arm at the end of the i^{th} round is less than $\left(\frac{1-p}{p}\right)^i$. Thus, the expected regret incurred during this phase up until the end of the ℓ^{th} round is bounded by $\sum_{i=1}^{\ell}\left(\frac{1-p}{p}\right)^i \times \beta^i \leq \ell$.

Overall, this implies that the strong expected regret up to time T (recall that

T is in round ℓ) is bounded by

$$\begin{aligned}&\left[\frac{2p^3}{(2p-1)^6}N(\log(N)+1)+\ell+\frac{(N-1)\ell}{2p-1}\right]\\ \leq&\left[\frac{2p^3}{(2p-1)^6}N(\log(N)+1)+\frac{N\log_\beta(T(\beta-1))}{2p-1}\right].\end{aligned}$$

Thus, the expected strong regret up to time T is $O(N\log(T)+N\log(N))$. □

Theorem 5. Under the Condorcet winner setting and for $1 < \beta < \frac{p}{1-p}$, the expected cumulative strong regret for WS-S is bounded by $\left[\frac{N^2p}{(2p-1)^2}+\frac{N\log(T(\beta-1))}{(2p-1)\log(\beta)}\right]$.

Proof. The proof mirrors that of Theorem 4, with the only difference being that we bound the length of exploration with a worse incumbent using the proof of Theorem 3 rather than Theorem 2, and the bound is $O(N^2)$. Due to its similarity, the proof is omitted. □

These results provide guidance on the choice of β. If β is too close to 1, then we spend most of the time in the exploration phase, which is guaranteed to generate strong regret. The last inequality in the proof of Theorem 4 suggests that asymptotic regret will be smallest if we choose β as large as possible without going beyond the $p/(1-p)$ threshold. Indeed, if β is too large, then WS-S may incur large regret in early exploitation stages when we have finished only a few rounds of exploration. In our numerical experiments we set $\beta = 1.1$, which satisfies the $p/(1-p)$ constraint assumed by our theory if $p > \beta/(1+\beta) \approx .524$. With a properly chosen β, the numerical experiments in section 3.5.2 suggest WS-S performs better than previously devised algorithms. At the same time, the best choice of β is dependent on p. Modifying WS-S to eliminate parameters that must be chosen with knowledge of p is left for future work.

Our regret bound grows as p, which is the minimal gap between two arms,

shrinks, and p tends to decrease as the number of arms N increases. Other dueling bandit algorithm for strong regret, such as RUCB and RMED, have regret bounds with better dependence on the gaps between arms. Modifying WS-S to provide improved dependence on these gaps is also left for future work.

3.4.4 Extension to Utility-Based Regret

We now briefly discuss utility-based extensions of weak and strong regret for the total order setting, following utility-based bandits studied in Ailon et al. (2014). Our regret bounds also apply here, with a small modification.

Suppose that the user has a utility u_i associated with each arm i. Without loss of generality, we assume $u_1 > u_2 > \cdots > u_N$, and as in the total order setting, we require that $p_{i,j} > 0.5$ when $i < j$. Typically the $p_{i,j}$ would come from the utilities of arms i and j via a generative model. We give an example in our numerical experiments.

Then, the single-period *utility-based weak regret* is $r(t) = u_1 - \max\{u_{i_t}, u_{j_t}\}$, which is the difference in utility between the best arm overall and the best arm that the user can choose from those offered. The single-period *utility-based strong regret* is $r(t) = u_1 - \frac{u_{i_t} + u_{j_t}}{2}$. To get zero regret under strong regret, the best arm must be pulled twice.

Our results from Section 3.4 carry through to this more general regret setting. Let $R = u_1 - u_N$ be the maximum single-period regret. Then, the expected cumulative utility-based weak regret for WS-W is $O\left(R\frac{N\log(N)}{(2p-1)^5}\right)$, and the expected cumulative utility-based strong regret for WS-S is $O(R\left[N\log(T) + N\log(N)\right])$.

3.5 Numerical Experiments

In this section, we evaluate WS under both the weak and strong regret settings, considering both their original (binary) and utility-based versions. In the weak regret setting, we compare WS-W with RUCB and QSA. In the strong regret setting, we compare WS-S with 7 benchmarks including RUCB and Relative Minimum Empirical Divergence (RMED) by Komiyama et al. (2015). We also include an experiment violating the total order assumption in Section 11 in the supplement. WS outperforms all benchmarks tested in these numerical experiments.

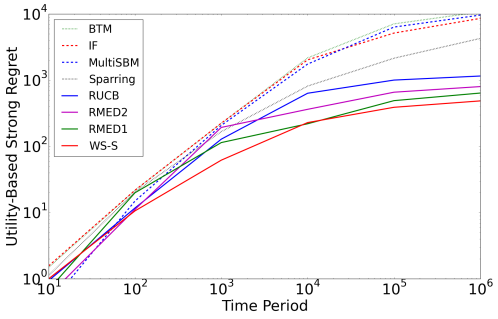
3.5.1 Weak Regret

We now compare WS-W with QSA and RUCB using simulated data and the Yelp academic dataset (Yelp, 2012).

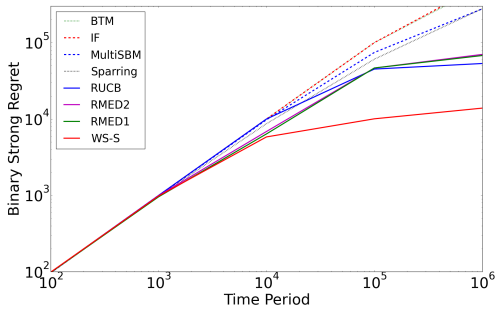
Simulated Data

In this example, we compare WS-W with RUCB and QSA on a problem with 50 arms and binary weak regret. Each arm is a 20-dimensional vector uniformly generated from the unit circle. We assume $p_{i,j} = 0.8$ for all $i < j$.

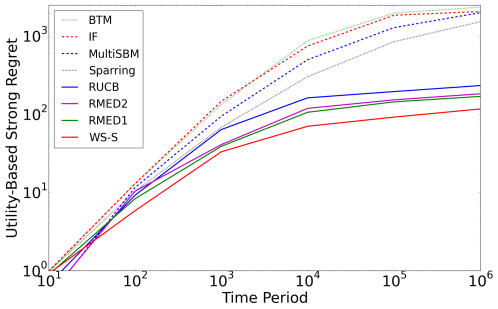
The results are summarized in Figure 3.2a. RUCB has approximately linear regret over the time horizon pictured. This is common in the dueling bandits literature, where many algorithms require $\sim 10^4$ comparisons before they achieve $\log(T)$ cumulative regret for 50 arms. WS-W finds the optimal arm



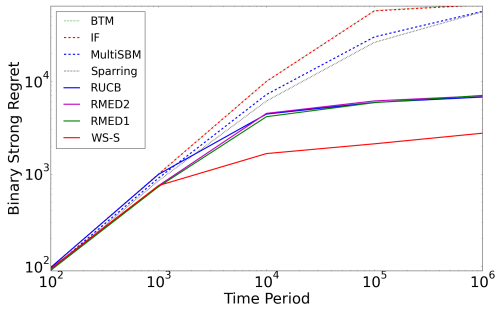
(a) Sushi dataset with utility-based strong regret



(b) Sushi dataset with binary strong regret



(c) MSLR dataset with utility-based strong regret



(d) MSLR dataset with binary strong regret

Figure 3.3: Comparison of the strong regret between WS-S and 7 benchmarks on the sushi and MSLR datasets. For utility-based strong regret, we start our plot from $t = 10$ since the performance of all algorithms are close to each other before $t = 10$. For the same reason, we start our plot from $t = 100$ for the binary strong regret. WS-S outperforms all benchmarks in all settings studied.

after $\sim$500 comparisons and has a regret that is consistent with our theoretically established constant expected cumulative weak regret.

Yelp Academic Dataset

In this example, we compare WS-W with RUCB and QSA using the Yelp academic dataset (Yelp, 2012) and utility-based weak regret.

We choose 100 restaurants from Las Vegas as our arms. Associated with each arm (restaurant) i is a 20-dimensional feature vector A_i, calculated using doc2vec

(Rehurek & Sojka, 2010) from its reviews. We select 49 users who have reviewed at least 20 of these 100 restaurants. For each user, we model their utility for restaurant i as $u_i = A_i \cdot \theta$, where θ is a 20-dimensional vector of preferences. We infer θ for each user using linear regression.

To model $p_{i,j}$, we then use the probit model. We let $\hat{\sigma}^2$ be the estimated variance of the residuals from the linear regression above. When presented with two restaurants, we model the user as taking independent random draws from a normal distribution with means u_i and u_j respectively and variances $\hat{\sigma}^2$, and choosing the restaurant with the larger draw. This gives $p_{ij} = \Phi(u_i - u_j)$, where $\Phi(\cdot)$ is the cdf for the normal distribution with mean 0 and variance $2\hat{\sigma}^2$.

We simulate performance for each user separately, and then average the results. These results are summarized in Figure 3.2b. WS-W outperforms RUCB and QSA, finding the optimal restaurant after ~ 500 iterations.

3.5.2 Strong Regret

In this section, we compare WS-S using binary and utility-based strong regret with 7 benchmarks from the literature. We use the sushi and MSLR datasets, which were previously used by Komiyama et al. (2016) and Zoghi et al. (2015) respectively to evaluate dueling bandit algorithms.

The sushi dataset (Komiyama et al., 2016) contains 16 arms corresponding to types of sushi, with pairwise preferences inferred from data on sushi preferences from 5000 users in Kamishima (2003). The MSLR dataset has 5 arms, corresponding to ranking algorithms, with pairwise preferences provided

in Zoghi et al. (2015). We give preference matrices $(p_{i,j})$ for both datasets in the supplement. For utility-based regret, we define $u_i = 2(1 - p_{1,i})$.

WS-S has a user-defined parameter β. In our experiments we set $\beta = 1.1$. The corresponding minimum p for which our theoretical bounds hold is $\beta/(1+\beta) \approx 0.52$. We recommend $\beta \approx 1.1$ for problems of 20 arms or fewer, and β closer to 1 for those problems with more arms that are likely to have p closer to $1/2$. We also conduct a sensitivity analysis of β in the supplement.

Figure 3.3 shows the results of our comparisons. WS-S outperforms all 7 benchmarks considered on both datasets using both variants of strong regret.

3.6 Conclusion

In this chapter, we consider dueling bandits for online content recommendation using both weak and strong regret. We propose a new algorithm, WS, with variants designed for the weak regret (WS-W) and strong regret (WS-S) settings. We prove WS has constant weak regret and optimal strong regret in T. In numerical experiments, WS outperforms all benchmarks considered on both simulated and real datasets.

CHAPTER 4

DUELING BANDITS WITH DEPENDENT ARMS

4.1 Introduction

In this chapter, we study dueling bandits in a different setting. We consider dueling bandits with utility-based weak regret, in the total order setting, when the total order is induced by a utility which is in turn a function of observable arm features, an unknown latent preference vector, and a known utility function. This framework includes the commonly used logit or Bradley-Terry (Revelt & Train, 1998; Yue et al., 2012) and probit models (Franses & Montgomery, 2002). We provide an algorithm, Comparing with the Best (CTB) that has expected cumulative utility-based weak regret that is constant in T, and that leverages the dependence between preferences over arms induced by the arm features and utility function to provide excellent empirical performance when prior information is available. While our regret bound's dependence on N is looser than Chen & Frazier (2017) (our dependence is 2^N in the worst case, and is N^{2d} when the utility function is linear over a d-dimensional space of preferences and arm features), our algorithm is more flexible in its ability to problem structure induced by the feature vectors, and outperforms it empirically by a substantial margin when N is small enough to allow computation that fully takes advantage of this problem structure.

Our exploitation of arm features is similar in spirit to work in the traditional (cardinal) multi-armed bandit setting on linear bandits (Rusmevichientong & Tsitsiklis, 2010; Abbasi-Yadkori et al., 2011).

The chapter is structured as follows. In section 4.2, we formulate our problem. In section 4.3, we introduce *Comparing The Best* (CTB) which we show in section 4.4 has CTB constant expected cumulative regret. In section 4.5, we discuss a efficient implementation method for a specific class of prior information. In section 4.6, we provide a Bayesian interpretation for CTB. In section 4.7, we compare CTB with three benchmarks using simulated datasets, in which CTB outperforms all benchmarks considered.

4.2 Problem Formulation

There are $N \geq 2$ arms, and each arm i has an observable and distinct d-dimensional feature vector A_i. Preferences between pairs of arms i, j are described by fixed but unknown probabilities $p_{i,j}$, where $p_{i,j} = 1 - p_{j,i}$ and $p_{i,j} \neq 0.5$ when $i \neq j$. We denote $p = \min_{i<j} \max(p_{i,j}, p_{j,i})$. By construction, $p > 0.5$.

At each time t, we pull two arms $X_{t,0}$ and $X_{t,1}$ (this act is called a "duel") and we observe feedback $Y_t \in \{0, 1\}$ indicating the winning arm: $Y_t = 0$ indicates arm $X_{t,0}$ won and $Y_t = 1$ indicates arm $X_{t,1}$ won. Conditioned on the arms pulled and the history (the arms pulled and the identity of the winner at times $t' < t$), Y_t is equal to 0 with probability $p_{i,j}$.

We suppose that the arms have a total order, i.e., that there exists an ordering of the arms such that $p_{i,j} > 0.5$ if and only if arm i is before arm j in this order. Moreover, we suppose this ordering is determined by a utility associated with each arm, $u(\theta, A_i)$, where u is a known utility function and $\theta \in \mathbb{R}^{d'}$ is an unknown preference vector. In particular, $p_{i,j} > 0.5$ if and only if $u(\theta, A_i) > u(\theta, A_j)$. The assumption that the total order be determined by $u(\theta, A_i)$ is without loss of

generality if we are willing to select d' to be sufficiently large and u to allow sufficient flexibility, although one may also choose a smaller d' and a less flexible u with the goal of obtaining smaller regret (described below) when these more restrictive modeling assumptions hold. We assume without loss of generality that the indices correspond to their ordering by utility, so $u(\theta, A_1) > u(\theta, A_2) > \cdots > u(\theta, A_N)$.

Several commonly used discrete choice models fall within this framework. For example, our framework includes the logit or Bradley-Terry model (Revelt & Train, 1998; Yue et al., 2012), in which $d' = d$, the utility function is $u(\theta, A_i) = \theta \cdot A_i$ and $p_{i,j} = \frac{\exp(u(\theta,A_i))}{\exp(u(\theta,A_i)+u(\theta,A_j))}$. Our framework also includes the probit model (Franses & Montgomery, 2002) in which $d' = d$ and the utility function is the inner product as with the logit model, but $p_{i,j} = \Phi(u(\theta, A_i) - u(\theta, A_j))$ where $\Phi(\cdot)$ is the standard normal cdf.

We define the utility-based weak regret $r(t)$ (henceforce referred to simply as the regret) at time t as $r(t) = u(\theta, A_1) - \max\{u(\theta, A_{X_{t,0}}), u(\theta, A_{X_{t,1}})\}$, which is the difference in utility between the best arm overall and the best arm available to the user from those offered. The cumulative regret up to time T is $R(T) = \sum_{t=1}^{T} r(t)$. We measure the quality of an algorithm by its expected cumulative regret.

We now develop an algorithm CTB, and show it has constant expected cumulative regret.

4.3 The *Comparing The Best* (CTB) Algorithm

In this section we propose an algorithm *Comparing The Best* (CTB) for this problem setting. This algorithm is based on the idea of "cells", which correspond to possible orderings of the arms by utility. It maintains a score for each cell, either explicitly or implicitly, which it initializes using optional prior information, and updates with the results from each duel.

We present a general version of CTB in this section that admits any prior information and explicitly maintains a score for each cell. Because the number of cells is exponential in the number of arms, explicitly maintaining scores for each cell is computationally infeasible for large problems. Thus, after presenting our theoretical results for the general CTB algorithm in section 4.4, we present a computationally efficient implementation of our algorithm in section 4.5 that can be used when the prior information can be expressed in terms of an initial score for each pair of arms. Although we present our algorithm in a frequentist setting, we show in section 4.6 that the scores used for each cell correspond to a Bayesian posterior on the value of θ, and CTB has a natural Bayesian interpretation.

To define CTB, we first define some terminology and notation: *winning spaces*, *cells*, a *score*, and the best arm corresponding to a cell. We begin with winning spaces.

Definition 4.3.1. Each pair of arms i, j defines a *winning space* $H_{i,j} := \{X \in \mathbb{R}^d : u(X, A_i) \geq u(X, A_j)\}$.

When $\theta \in H_{i,j}$, arm i is preferred over arm j. We use the phrases "arm A_i wins over arm A_j in a duel", and "winning space $H_{i,j}$ wins the duel" interchangeably.

Each pair of arm determines two winning spaces and all winning spaces partition the space $\mathbb{R}^d$ into cells, where each cell is an intersection of winning spaces. To define notation to support working with cells, we first define $H_{i,j}(k) = H_{i,j}$ when $k = 0$ and $H_{i,j}(k) = H_{j,i}$ when $k = 1$. For a binary vector V, we let $V[k]$ denote the k^{th} element of V. Then, we have the following definition.

Definition 4.3.2. The *cell* C corresponding to a length $\frac{N(N-1)}{2}$ binary vector V is

$$C(V) := \cap_{i<j} H_{i,j}\left(V\left[\frac{1}{2}(2N-i)(i-1)+j-i\right]\right).$$

We assign binary vectors indexing cells, all of length $\frac{N(N-1)}{2}$, to integers lexicographically. Let V_k denote the k^{th} such binary vector, let $M = 2^N$ denote the number of cells, and let $C_i = C(V_i)$. With this definition, $C_1 = C(V_1) = C([0, 0, \cdots, 0])$ and thus $C_1 = \cap_{i<j} H_{i,j}$ and $\theta \in C_1$. Some cells C_i may be empty. We call these *empty cells*. Let $J_k = \{(i, j) | C_k \subseteq H_{i,j}\}$, which is the collection of indices of the winning spaces that contains C_k.

Figure 4.1 illustrates winning spaces and cells.

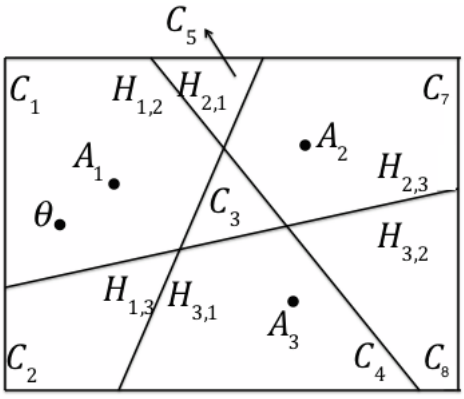


Figure 4.1: Illustration of winning spaces and cells. The index of the cell and its corresponding binary vectors are: C_1 and $(0, 0, 0)$; C_2 and $(0, 0, 1)$; C_3 and $(0, 1, 0)$; C_4 and $(0, 1, 1)$; C_5 and $(1, 0, 0)$; C_7 and $(1, 1, 0)$; C_8 and $(1, 1, 1)$. In this case, cell C_6 is an empty cell since the intersection of $H_{2,1}$, $H_{1,3}$ and $H_{3,2}$ is empty.

We define a score $m_i(t)$ associated with each cell C_i at time t. Later in section 4.6 we will interpret this score as a monotone transformation of the posterior probability that θ is in this cell. This score will be initialized to some value $m_i(0)$, discussed below, and then will be incremented each time a winning space containing C_i wins a duel. That is,

$$m_i(t) = m_i(0) + \sum_{k=1}^{t} \mathbb{1}\{C_i \subseteq H_{X_{k,1},X_{k,2}}(Y_k)\}. \tag{4.1}$$

Each cell C_i assigns a preference order to the arms. Let $B(i)$ be the arm that would be best if θ were in C_i. More formally, $B(i)$ is the unique j such that $C_i \subseteq H_{j,k}$, $\forall k \neq j$. Since $\theta \in C_1$, we know $B(1) = 1$.

With this notation, we now define the *Comparing The Best* (CTB) algorithm in Algorithm 6. CTB pulls the arm that is best according to the cell with the highest score $m_i(t)$, and the arm that is best according to the cell with the highest score among those that have different best arm from the first arm chosen. If we interpret $m_i(t)$ as being a monotone transformation of the posterior probability that $\theta \in C_i$, then we are selecting arms by selecting two cells that have different best arms, and are together most likely to contain θ.

Algorithm 6 *Comparing The Best* (CTB)

for t ≤ T **do**
  Step 1: Pick $X_{t,0} = B(\arg\max_i m_i(t))$, breaking ties arbitrarily
  Step 2: Pick $X_{t,1} = B\left(\arg\max_{i:B(i)\neq X_{t,0}} m_i(t)\right)$, breaking ties arbitrarily
  Step 3: Observe the noisy feedback Y_t and update $m_i(t)$ using Equation (4.1)
  Step 4: t=t+1
end for

Choice of $m_i(0)$: Here we offer guidance on the choice of $m_i(0)$, which is left general in the description of CTB to allow the user the flexibility to influence the arms pulled with prior information about the value of θ, and to trade off regret against CTB's computational performance. In doing so, there are four considerations:

First, by setting $m_i(0)$ larger for those cells that the user believes are more likely to contain θ, the user encourages CTB to select those cells more often. If the user correctly sets $m_i(0)$ larger for the cell that contains θ, this tends to pull the best arm more often and decrease regret. We show in section 4.6 that $m_i(0)$ can be interpreted in terms of the prior probability that $\theta \in C_i$, and one can leverage this relationship to convert prior information on θ into values for $m_i(0)$.

Second, by setting $m_i(0)$ to be $-\infty$ for those cells that user is certain do not contain θ, she can lead CTB to never select those cells. One may safely do this for empty cells, in which model assumptions imply θ cannot reside. Doing this for other cells is dangerous, as setting cell $m_1(0)$ to $-\infty$ can cause CTB to have linear regret.

Third, in the absence of prior information, one may simply set $m_i(0) = 0$ for all cells that may contain θ. We show in the next section show that as long as $m_1(0) > -\infty$, the expected cumulative regret is finite.

Fourth, there is a computational aspect to setting $m_i(0)$. We show below in section 4.5 that if each $m_i(0)$ can be written as a sum across pairs of arms of a score associated with each pair, then we can implement CTB in a computationally efficient manner that scales to many arms. In contrast, if one sets $m_i(0)$ without enforcing structure, the computation required to implement

Algorithm 6 grows exponentially with the number of arms.

With these considerations in mind, we propose 3 specific ways to set $m_i(0)$, and evaluate them in numerical experiments:

- For situations with loose computational requirements or few arms, and no prior information, we recommend setting $m_i = 0$ for all non-empty cells and $m_i = -\infty$ for all empty cells. We call this CTB−1.
- For situations with strict computational requirements and no prior information, we recommend setting $m_i = 0$ for all cells. Then CTB can be implemented using the efficient method described in section 4.5. We call this CTB−2.
- For situations with loose computational requirements or few arms, and strong prior information, we recommend setting m_i from the prior according to the method described in section 4.6. We call this CTB−3.

4.4 Theoretical Results

In this section, we prove the expected cumulative regret of CTB is bounded by a constant. The main idea behind our proof is to show that for each cell C_i with $B(i) \neq 1$, $\mathbb{E}[\sum_{t=0}^{\infty} 1\{m_i(t) \geq m_1(t)\}]$ is bounded by a constant. We show this in turn by relating $m_1(t) - m_i(t)$ to a random walk with a larger probability of increasing than of decreasing. The following lemma, whose proof is in the supplement, allows us to bound the number of times this stochastic process takes values less a constant.

Lemma 6. Let $p \in (0.5, 1]$. Suppose $Z(t)$ is a stochastic process with filtration $\mathcal{F}_t$,

$Z(0) = 0$ and $P(Z(t+1) = Z(t)+1|\mathcal{F}_t) \geq p$, then we have $\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq S\}\right] \leq \frac{p+S(2p-1)}{(2p-1)^2}$ for $S \in \mathbb{N}$.

We now proceed with the larger proof by defining

$$q_{i,j}(t) = \sum_{k=1}^{t} \mathbb{1}\{X_{k,0} = i, X_{k,1} = j, Y_k = 0\} + \sum_{k=1}^{t} \mathbb{1}\{X_{k,0} = j, X_{k,1} = i, Y_k = 1\}, \tag{4.2}$$

which is the number of times up to time t that arm i beats arm j in a duel. Then we can rewrite $m_i(t)$ in terms of $q_{i,j}(t)$ as,

$$m_k(t) = m_k(0) + \sum_{(i,j)\in J_k} q_{i,j}(t). \tag{4.3}$$

The definition of C_1 implies $J_1 = \{(i,j), \forall i < j\}$ and $m_1(t) = m_1(0) + \sum_{i<j} q_{i,j}(t)$. Let $N_{i,j}(t) = q_{i,j}(t) + q_{j,i}(t)$ denote the number of times we have pulled arms i and j. The next lemma shows $\mathbb{E}[N_{i,j}(t)]$ is bounded by a constant for $1 < i < j$.

Lemma 7. For $1 < i < j$, if $m_1(0) > -\infty$, we have $\mathbb{E}[N_{i,j}(t)] \leq M' \frac{p-\Delta(2p-1)}{(2p-1)^2}$, where M' is the number of cells i with $m_i(0) > -\infty$, and $\Delta = \min_{s=1,\cdots M}\{m_1(0) - m_s(0)\} \leq 0$.

Proof. Let $1 < i < j$. Let $D_{i,j}(t)$ be an indicator function equal to 1 if and only if we pull arms i and j at time t. Given that we pull arm i, we can only also pull arm j when there is a cell C_s under which j is the best arm and for which $m_s(t) \geq m_1(t)$. Moreover, under the assumption that $m_1(0) > -\infty$, $m_s(t) \geq m_1(t)$ is only possible if $m_s(0) > -\infty$. Thus, $D_{i,j}(t) = 1$ implies $\max_{s:B(s)=j, m_s(0)>-\infty} m_s(t) \geq m_1(t)$. Adopting the convention here and in the rest of the proof that maxima and sums over sets

of cells are taken only over those cells with $m_s(0) > -\infty$, we have

$$\begin{aligned}
D_{i,j}(t) &= D_{i,j}(t) \cdot \mathbb{1}\left\{\max_{s:B(s)=j} m_s(t) \geq m_1(t)\right\} \\
&\leq D_{i,j}(t) \sum_{s:B(s)=j} \mathbb{1}\{m_s(t) \geq m_1(t)\} \\
&= D_{i,j}(t) \sum_{s:B(s)=j} \mathbb{1}\left\{\sum_{(i',j')\in J_s} q_{i',j'}(t) + m_s(0) \geq \sum_{(i',j')\in J_1} q_{i',j'}(t) + m_1(0)\right\} \\
&= D_{i,j}(t) \sum_{s:B(s)=j} \mathbb{1}\left\{\sum_{(i',j')\in J_s\setminus J_1} q_{i',j'}(t) + m_s(0) \geq \sum_{(i',j')\in J_1\setminus J_s} q_{i',j'}(t) + m_1(0)\right\} \\
&= D_{i,j}(t) \sum_{s:B(s)=j} \mathbb{1}\left\{\sum_{(i',j')\in J_s\setminus J_1} q_{i',j'}(t) - q_{j',i'}(t) \geq m_1(0) - m_s(0)\right\} \\
&\leq D_{i,j}(t) \sum_{s:B(s)=j} \mathbb{1}\left\{\sum_{(i',j')\in J_s\setminus J_1} q_{i',j'}(t) - q_{j',i'}(t) \geq \Delta\right\},
\end{aligned}$$

where the fourth equation holds because J_s has the property that $(i', j') \in J_s \iff (j', i') \notin J_s$, and similarly for J_1. Thus, $(i', j') \in J_s \setminus J_1 \iff i', j' \in J_s$ and $i', j' \notin J_1 \iff j', i' \notin J_s$ and $j', i' \in J_1 \iff (j', i') \in J_1 \setminus J_s$.

Thus, we have

$$N_{i,j}(t) = \sum_{k=1}^{t} D_{i,j}(k) \leq D_{i,j}(k) \sum_{s:B(s)=j} \sum_{k=1}^{t} \mathbb{1}\left\{\sum_{(i',j')\in J_s\setminus J_1} q_{i',j'}(k) - q_{j',i'}(k) \geq \Delta\right\}.$$

Fix an s with $B(s) = j$ and let $Z(k) = \sum_{(i',j')\in J_s\setminus J_1} q_{i',j'}(k) - q_{j',i'}(k)$, so that

$$N_{i,j}(t) \leq \sum_{s:B(s)=j} \sum_{k=1}^{t} D_{i,j}(k) \cdot \mathbb{1}\{Z(k) \geq \Delta\}.$$

We observe that $Z(k)$ is like a random walk, except that changes in only some time periods. We now describe the conditional distribution of $Z(k+1)$ given the history up to time k. Later, we will refer to the σ-algebra generated by this history as $mathcalH_k$.

- If the arms $X_{k,0}$, $X_{k,1}$ that we pull satisfy $(X_{k,0}, X_{k,1}) \in J_s \setminus J_1$, then $Z(k+1) \in \{Z(k)-1, Z(k)+1\}$ and the conditional probability that $Z(k+1) = Z(k)-1$ is $p_{X_{k,1},X_{k,0}} \geq p$. This lower bound holds because $(X_{k,0}, X_{k,1}) \notin J_1$ implies $X_{k,1} < X_{k,0}$.
- Similarly, if $(X_{k,1}, X_{k,0}) \in J_s \setminus J_1$, then $Z(k+1) \in \{Z(k)-1, Z(k)+1\}$ as before, and the conditional probability that $Z(k+1) = Z(k)-1$ is $p_{X_{k,0},X_{k,1}} \geq p$, because $(X_{k,1}, X_{k,0}) \notin J_1$ implies $X_{k,0} < X_{k,1}$.
- Otherwise, if neither $(X_{k,0}, X_{k,1})$ nor $(X_{k,1}, X_{k,0})$ is in $J_s \setminus J_1$, then $Z(k+1) = Z(k)$.
- The definition of J_1 prevents having both $(X_{k,0}, X_{k,1})$ and $(X_{k,1}, X_{k,0})$ in $J_s \setminus J_1$.

When $D_{i,j}(k) = 1$, so that we pull arms i and j (either $X_{t,0} = i$ and $X_{t,1} = j$ or vice versa) we will be in one of the first two cases, because $B(s) = j$ implies cell s considers j to be the best arm, and so $(j,i) \in J_s$, and $i < j$ implies $(j,i) \notin J_1$. Thus, $D_{i,j}(k) = 1$ implies $Z(k+1) \neq Z(k)$, and we have

$$N_{i,j}(t) \leq \sum_{s:B(s)=j} \sum_{k=1}^{t} \mathbb{1}\{Z(k+1) \neq Z(k), Z(k) \geq \Delta\}.$$

We will perform a random time change to study the dynamics over only those time periods where $Z(k)$ changes. Define $\tau_0 = 0$, $\tau_m = \min_k\{k > \tau_{m-1}, Z(k) \neq Z(k+1)\}$. Because the event $Z(k) \neq Z(k+1)$ is measurable given the history at time k, $\mathcal{H}_k$, as described in the dynamics of $Z(\cdot)$ above, each τ_m is a stopping time. Define $\zeta = \inf\{m : \tau_m = \infty\}$, which is the lifetime of the random change of time. We have,

$$N_{i,j}(t) \leq \sum_{s:B(s)=j} \sum_{m=1}^{\zeta-1} \mathbb{1}\{Z(\tau_m) \geq \Delta\}. \tag{4.4}$$

We let $W(m) = Z(\tau_m)$ for $m < \zeta$ (i.e., m with $\tau_m < \infty$), and $W(m) = W(m-1) + \epsilon_m$ for $m \geq \zeta$, where ϵ_m are iid random variables taking value -1 with probability

p and value 1 with probability $1-p$. Observe that ζ is measurable with respect to $\mathcal{H}_\infty$, so that the event $m < \zeta$ is measurable with respect to $\mathcal{H}_{\tau_m}$. We define an augmented filtration, letting $\mathcal{F}_m$ to be the σ-algebra generated by $\mathcal{H}_{\tau_{\min(m,\zeta)}}$ and $(\epsilon_{m'} : m' \leq m)$. With this construction, $W(m+1) - W(m) \in \{-1, +1\}$ and $P(W(m+1) = W(m) - 1 | \mathcal{F}_m) \geq p$. Thus, by Lemma 1,

$$\sum_{m=1}^{\zeta} \mathbb{1}\{Z(\tau_m) \geq \Delta\} = \sum_{m=1}^{\zeta} \mathbb{1}\{W(m) \geq \Delta\} \leq \sum_{m=1}^{\infty} \mathbb{1}\{W(m) \geq \Delta\} \leq \frac{p - \Delta(2p-1)}{(2p-1)^2}.$$

Combining this with (4.4) and using the fact that the number of cells with $m_s(0) > -\infty$, M', bounds the sum over s, we obtain our result. □

Based on Lemma 7 and a union bound, we obtain our main theorem:

Theorem 6. Let $\Lambda = u(\theta, A_1) - u(\theta, A_N)$. If $m_1(0) > -\infty$, CTB's expected cumulative regret is bounded by $\frac{(N-1)(N-2)}{2} M' \frac{p-\Delta(2p-1)}{(2p-1)^2} \Lambda$.

In general, M' can be as large as 2^N. However, as discussed above, we may set $m_i(0) = -\infty$ for all the empty cells and assign finite $m_i(0)$ to empty cells (CTB−1). In this setting, since each cell assigns a ranking over arms and different cells give different rankings, we can bound M' by the number of permutations of N arms, $N!$. Moreover, when the utility function is linear and $d' = d$, results in Jamieson & Nowak (2011) show M' is $O(N^{2d'})$.

4.5 Computation for Decomposable m_i

CTB achieves a constant expected cumulative regret. However, a naive implementation of Algorithm 6 requires a great deal of memory to store $m_i(t)$

for each cell, which makes it computationally challenging for problems with many arms. In this section, we consider a special case of CTB where $m_i(0)$ can be expressed in terms of an initial score for each pair of arms. Specifically, we suppose that there exists a $r_{i,j}$ such that

$$m_k(0) = \sum_{(i,j)\in J_k} r_{i,j} \quad \forall k. \tag{4.5}$$

Here $r_{i,j}$ can be interpreted as a prior indicating the extent to which we believe that arm i is preferred over arm j. In this special case, we describe an efficient computation method that scales to problems with many arms.

Instead of storing $m_i(t)$, this method stores $r_{i,j}$ and $q_{i,j}(t)$ and uses them to reconstruct $m_i(t)$ with Equation 4.3. Then, Steps 1 and 2 in Algorithm 6 are written as optimization problems in which $m_i(t)$ is replaced by this expression in terms of $q_{i,j}(t)$ and $r_{i,j}$. Toward this end, let $e_{i,j}$ denote a binary variable that will take value $e_{i,j} = 1$ if we are to select a cell in $H_{i,j}$ and 0 otherwise. Then, based on Equation 4.3, maximizing $m_i(t)$ is equivalent to maximizing $\sum_{i,j:i\neq j} e_{i,j} \times (q_{i,j}(t) + r_{i,j})$.

To find the best arm suggested by $\arg\max_i m_i(t)$ in Step 1, and suggested by a similar argmax in Step 2, it is sufficient to find $\max_{i:B(i)=k} m_i(t)$ for each arm k. This is the cell with largest $m_i(t)$ among those that believe k is best. This problem is:

$$\begin{aligned} \text{maximize} \sum_{i,j:i\neq j} & e_{i,j} \times (q_{i,j}(t) + r_{i,j}) \\ \text{subject to } & e_{k,j} = 1, \ \forall j \neq k \\ & e_{i,j} + e_{j,i} = 1, \ i, j = 1, ..., N, i \neq j \\ & e_{i,j} \in \{0, 1\}, \ \forall i \neq j \end{aligned} \tag{4.6}$$

There are three conditions in Equation 4.6. The first condition is $e_{k,j} = 1$

$\forall j \neq k$, which means cell C_ℓ that satisfies the first condition must lie in the winning space $H_{k,j}$, $\forall j \neq k$. In other words, C_ℓ ranks arm A_k better than any others and thus $B(\ell) = k$. The second and third condition together guarantee that cell C_ℓ either belongs to $H_{i,j}$ or $H_{j,i}$.

Though Equation 4.6 is an integer linear programming problem, which are usually computationally challenging, it is in fact easy to solve: the maximum value of this problem is reached when $e_{i,j} = 1$ if $r_{i,j} + q_{i,j}(t) > q_{j,i}(t) + r_{j,i}$ for all $i \neq j$, $e_{i,j} = 0$ if this strict inequality is reversed, and breaking ties arbitrarily between the solutions $(e_{i,j} = 1, e_{i,j} = 0)$ and $(e_{i,j} = 0, e_{i,j} = 1)$ for those i, j with equality.

Denote the maximum value of this problem at time t as $f(k, t)$. After knowing $f(k, t) = \max_{B(i)=k} m_i(t)$, finding the arm with largest $m_i(t)$ in Step 1 is equivalent to finding $\arg\max_k f(k, t)$. Finding the arm with large $m_i(t)$ among those with a different best arm than $X_{t,0}$ in Step 2 is equivalent to finding $\arg\max_{k \neq X_{t,0}} f(k, t)$.

For general values of $m_i(0)$ that do not satisfy (4.5), finding the largest $m_i(t)$ is computationally challenging. However, in applications, instead of setting $m_i(0)$ directly, we may have some prior information about the probability that the user prefers arm i over arm j. This information can be used to construct $r_{i,j}$ since CTB guarantees constant regret regardless of the values that $m_i(0)$ take.

4.6 Bayesian Interpretation

Although our problem is formulated in a frequentist setting, we show here that CTB has a Bayesian interpretation. In this section, we construct a Bayesian

posterior on θ given a prior and given an assumption that $p_{i,j} = q > 0.5$ for all $i < j$, where q may be the same or different from p, and $p_{i,j}$ may or may not be constant across i, j in reality.

We put a prior distribution p_0 on θ, which induces a prior on the identity of the cell containing θ. The prior probability that θ is in cell i is written $p_0(C_i)$, and is obtained by integrating p_0 over C_i. Let $p_t(C_i)$ indicate the posterior probability that θ is in cell C_i, at time t, given $p_{i,j} = q$ for all $i < j$. The following pair of lemmas give recursive and non-recursive expressions for p_t.

Lemma 8. For compactness of notation, let $i = X_{t,0}$ and $j = X_{t,1}$. Then the posterior distribution p_{t+1} is,

$$p_{t+1}(x) = \begin{cases} \frac{p_t(x)q}{p_t(H_{i,j}(Y_k))q+(1-p_t(H_{i,j}(Y_k)))(1-q)} & \text{if } x \in H_{i,j}(Y_t) \\ \frac{p_t(x)(1-q)}{p_t(H_{i,j}(Y_k))q+(1-p_t(H_{i,j}(Y_k)))(1-q)} & \text{if } x \notin H_{i,j}(Y_t) \end{cases}$$

Based on this lemma, we can rewrite the posterior distribution in terms of $m_i(t) - m_i(0)$.

Lemma 9. For each cell C_i, the posterior distribution after t comparison is

$$p_t(C_i) \propto p_0(C_i) q^{m_i(t)-m_i(0)} (1-q)^{t-m_i(t)+m_i(0)}.$$

We leave the proof of both Lemmas to the appendix. Lemma 9 allows us to rewrite $p_t(C_i)$ as

$$\begin{aligned} p_t(C_i) &\propto p_0(C_i) q^{m_i(t)-m_i(0)} (1-q)^{t-m_i(t)+m_i(0)} \\ &\propto p_0(C_i) (\frac{q}{1-q})^{m_i(t)-m_i(0)}. \end{aligned}$$

Thus, choosing the cell to maximize the posterior probability is equivalent to choosing the cell to maximize $\log(p_0(C_i)) + (m_i(t) - m_i(0)) \log\left(\frac{q}{1-q}\right)$. Thus, if

$$m_i(0) = \log(p_0(C_i)) / \log\left(\frac{q}{1-q}\right), \tag{4.7}$$

then maximizing the posterior probability that θ is in C_i is equivalent to maximizing $m_i(t)$, the first cell selected by CTB is the cell with the largest posterior probability of containing θ, and the second cell selected is the largest among those with a different best arm from the first.

Thus, if one has prior information about the location of θ and an estimate q of a typical value of p_{ij}, then a natural way to set $m_i(0)$ is via (4.7). In addition, since $p_0(C_i) = 0$ for empty cells, following (4.7) also sets $m_i(0) = -\infty$ for these cells as discussed before.

4.7 Numerical Experiments

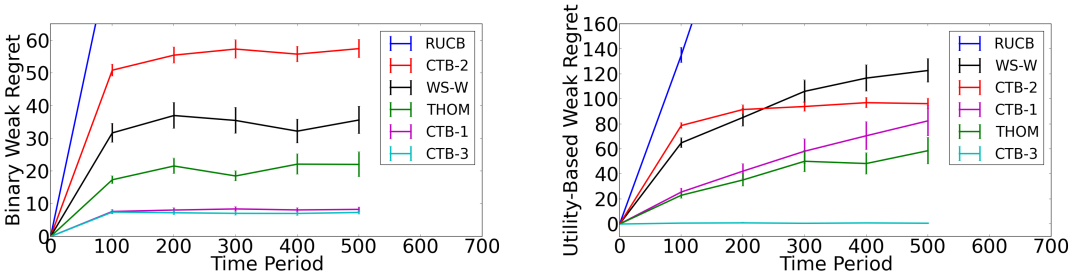
In this section, we compare the three variants of CTB described in section 4.3, CTB-1, CTB-2, and CTB-3, with three benchmarks: Thompson Sampling, Relative Upper Confidence Bound (RUCB) and Winner-Stays (WS).

- Thompson sampling uses a posterior distribution over θ computed by beginning with a prior distribution on the location of θ, and updating it using Bayes rule and knowledge of $p_{i,j}$. At time t, it generates θ_t from this posterior distribution p_t and pulls the two arms that θ_t ranks as best and second best. In our implementation, we track the prior/posterior explicitly by storing a probability for each cell. We emphasize that Thompson sampling as we consider it here requires knowledge of $p_{i,j}$ which is not typically not available.
- RUCB is as described in Zoghi et al. (2014). We choose it as our benchmark over other algorithms designed for strong regret from the

literature because it works well relative to other algorithms designed for strong regret in previous literature when a Condorcet winner exists, and existence of a Condorcet winner is a consequence of our total order assumption. Though there are algorithms that outperform RUCB in some settings such as CCB and SCB (Zoghi et al., 2015), they typically work better when a Condorcet winner does not exist.

- WS is as described in Chen & Frazier (2017), and is selected because it is designed for the weak regret setting. In our plots, WS-W is the variant of WS designed specifically for weak regret.

We consider two experimental settings described below, with results pictured in Figure 4.2.



(a) Binary Regret and Constant $p_{i,j}$

(b) Bradley-Terry Regret and $p_{i,j}$

Figure 4.2: Performance comparison of the three CTB variants from section 4.3 against benchmarks WS-W, RUCB and Thompson Sampling (THOM) using simulated datasets. CTB−3 and Thompson sampling use prior information, and in this group CTB−3 performs best. Among the four algorithms that do not use prior information, CTB−1 performs best. CTB−2 under-performs WS-W in the binary regret setting and for $t = 100, 200$ in the Bradley-Terry setting, and outperforms WS-W when $t = 300, 400, 500$ in the Bradley-Terry setting.

Since RUCB performs poorly in both experiments compared with other algorithms, we set the y-axis to emphasize the relative performance of the other algorithms. We include a plot over a wider y-axis showing RUCB's performance in the supplement.

4.7.1 Binary Regret and Constant $p_{i,j}$

In this experimental setting, we set $p_{i,j} = 0.8$ for all $i < j$. We have $N = 20$ arms uniformly generated from the 2-dimensional unit circle. The preference vector θ is generated uniformly at random from the 2-dimensional unit circle. We set regret to 1 if both of the pulled arms are not optimal, i.e. $u(\theta, A_1) = 1$ and $u(\theta, A_i) = 0$ for $i \neq 1$. To satisfy our previous assumption that $u(\theta, A_i)$ be distinct across i, we may equivalently set $u(\theta, A_i) = i \cdot \epsilon$, and take ϵ small.

Figure 4.2a shows that CTB−1 and CTB−3 perform comparably and both outperform WS-W and Thompson Sampling. CTB−2 does not perform as well as WS-W and Thompson Sampling. Both Thompson sampling and CTB−3 have access to the correct prior and use the true value of p to perform updating.

4.7.2 Bradley-Terry Regret and $p_{i,j}$

In this experimental setting, we set utility using the Bradley-Terry model described in section 4.2. As in the first experimental setting, we have $N = 20$ arms on the 2-dimensional unit circle. Among these arms, 19 are uniformly generated from $\{x < 0, y < 0, x^2 + y^2 = 1\}$ and 1 arm is uniformly generated from $\{x > 0, y > 0, x^2+y^2 = 1\}$. The user's preference θ is also uniformly generated from $\{x > 0, y > 0, x^2 + y^2 = 1\}$, but the Bayesian algorithms (CTB−3and Thompson sampling) use another less information prior: that θ is uniform on the unit circle. Thompson sampling performs its update using the true $p_{i,j}$, while CTB−3 uses a rough approximation of $q = 0.6$ to set $m_i(0)$ to model the fact that we would not know p or $p_{i,j}$ in practice.

Figure 4.2b shows that both CTB−3 and Thompson Sampling takes advantage of the prior information and the dependence among arms. CTB−3 uses this information more efficiently and significantly outperforms Thompson Sampling. Among the four algorithms (CTB−1, CTB−2, RUCB and WS) that do not use prior information, CTB−1 performs best. Though CTB−2 does not perform as well as WS at $t = 100, 200$, it outperforms WS when $t = 300, 400, 500$.

4.8 Conclusion

In this chapter, we consider dueling bandits for weak regret, with application to recommender systems and online content recommendation. We formulate a new setting which differs from the traditional dueling bandits in which arms are dependent. We propose an algorithm CTB, and show it has constant expected cumulative regret and strong empirical performance.

CHAPTER 5

INCENTIVIZING EXPLORATION WITH HETEROGENEOUS USER PREFERENCES

5.1 Introduction

In this chapter, we study attribute feedback in the setting of incentivizing exploration with heterogeneous agent preferences. In this problem, arms have unknown multivariate attributes, and agents have heterogeneous linear utility functions that map these attribute vectors onto utilities. Agents see noisy observations of attributes of arms pulled by all previous agents, and estimate an arms' attribute vector by the simple average of these observations. Agents are selfish, and pull the arm with the largest estimated utility summed with an optional arm-specific incentivizing payment chosen by the principle. We study strategies for choosing such incentive payments that seek to maximize the total utility derived by agents, subject to a limitation on the total incentive payment. To accomplish this goal, a strategy must induce sufficient exploration to reveal arms' attributes, while still letting agents select myopically and according to their preferences sufficiently often that high-utility arms are chosen and incentive payments are kept small.

Our problem setting models online review aggregators like Amazon, Yelp, and Tripadvisor that host crowdsourced reviews. Users of these websites wish to use the reviews hosted there to choose the product / restaurant / vacation (generically referred to as an "item") that is best according to their preferences. These reviews provide not just cardinal feedback, but also a description of items' attributes that a user may consider together with their personal preferences to

select their preferred item. An item with few reviews might have inaccurate attribute estimates, leading users to avoid it even though it may actually be their best choice. Without incentives, this situation may persist and decrease welfare for the platform's user base. By offering incentives, either through price reductions by Amazon or coupons from Yelp or Tripadvisor, a platform may induce more reviews of unexplored items and provide more value over the long term.

Our problem setting also applies to crowd science platforms like eBird (Frazier et al., 2014; Sullivan et al., 2009). EBird guides birding enthusiasts through a website to explore and report their findings to the birding community. Each user report contains information about when, where and how they go birding and what birds they see and hear. EBird may wish to incentivize enthusiasts to explore less-explored birding locations and provide more accurate reports on these locations. Each enthusiast may have different preference over a location's attributes such as the diversity of bird species, weather, distance and safety. By offering enthusiasts incentivies to explore, eBird can create a more accurate body of reports and provide better value to the birding community.

In this problem context, we study a simple policy that usually exploits, incentivizing agents to pull an arm only when the set of agent utility functions that would pull this arm without incentives has probability below a time-varying threshold. In our paper, we assume all arms are some agents' best arm. Under this assumption, we prove that with $O(N^2)$ payment budget, this policy has $O(N^2 + M(\log(T))^2)$ cumulative expected regret where M is an upper bound on the limiting marginal probability density of agent utilities that

are nearly indifferent between their best and the second best arm. If all agents' utility difference between their best and second best arm is bounded below by a positive number, which typically happens when the agent utility distribution is discrete, this policy achieve constant cumulative expected regret $O(N^2)$. The key difference between our problem setting and both the homogenous preference setting and the traditional multi-armed bandits setting is that we must incentivize agents to try suboptimal arms much less often, since all arms are some agents' best arm. Essentially, heterogeneity provides free exploration. These results suggest that heterogeneous agent preferences reduce but do not eliminate the need to incentive exploration, in relation to single-preference settings.

We broadly categorize the relevant previous literature into two categories based on whether there is money transfer. With money transfer, Frazier et al. (2014) considers a problem setting where the principal pays agents money to explore. This work assumes all agents have equal value for money and provide a complete characterization of achievable reward with a fixed budget. Han et al. (2015) generalizes this framework to include agents with heterogeneous value for money, and to allow an external signal to provide partial information about this valuation. Under this setting, this work proves a bound on achievable reward as a function of the budget and the signal scheme.

Without money transfer but using information asymmetry, Kremer et al. (2014) considers a simple model where agents arrive to the principal one by one and there are only two actions at each time. Mansour et al. (2015) generalizes Kremer et al. (2014) by allowing a finite number of actions at each time. Mansour et al. (2016) considers a problem setting where there are

multiple agents at each time and agents may interact with each other. In these papers, the principal provides each agent a recommendation at each time that is Bayesian incentive-compatible. They prove the principal can achieve constant regret when utilities are deterministic and logarithmic regret when utilities are stochastic.

We structure our paper as follows: Section 5.2 formulates our problem; Section 5.3 states our algorithm and proves that we can achieve $O(N^2 + M(\log(T))^2)$ regret with $O(N^2)$ incentive budget; Section 5.4 constructs an example showing regret is $\Omega(\log(T))$ in the worst case, regardless of incentive budget.

5.2 Problem Setting

We have N arms. Arm i has a fixed but unknown attribute vector $u_i \in \mathbb{R}^m$. A stream of myopic selfish agents come to our system. Agent t has linear preferences over attributes described by a vector $\theta_t \in \mathbb{R}^m$ that is unknown to the principal and drawn i.i.d. from a known distribution $F(\cdot)$ with compact support. We refer synonomously to an agent and that agent's preference vector: when we say "an agent θ", we mean "an agent with preference vector θ."

Each agent t chooses an arm to pull A_t, according to a process described below, and obtains utility $\theta_t \cdot u_{A_t}$. The principal and all agents then see a noisy observation of the attribute vector of the pulled arm of the form $O_t = u_{A_t} + \epsilon_t$, where $\epsilon_t \sim N(0, \sigma^2 I_m)$ is independent normally distributed noise, and I_m denotes an m-dimensional identity matrix. Although we assume a common variance across attributes for simplicity of presentation, our theoretical results hold if the

variance differs.

At each time t, for each arm i, we (the principal) offer a non-negative payment $c_{i,t} \geq 0$ based on previous observations. We assume that agent t chooses to pull the arm that myopically maximizes the sum of this payment and an estimate of the utility obtained $\theta_t \cdot u_{i,t}$ where $u_{i,t}$ denotes the simple average of O_s over all previous pulls of arm i. In this paper, we assume all arms have been pulled once at time $t = 0$ and $u_{i,0}$ denotes a random draw from the arm attribute vector. For $t > 0$, denote $u_{i,t} = \frac{\sum_{s<t} O_s 1\{A_s=i\}+u_{i,0}}{\sum_{s<t} 1\{A_s=i\}+1}$ and $A_t = \arg\max_i\{\theta_t \cdot u_{i,t} + c_{i,t}\}$, breaking ties in favor of the arm with the highest incentive. We use $c_t = c_{A_t,t}$ to denote the actual incentive payment at time t.

This behavior may be recovered if agents are Bayesian and share a common non-informative prior distribution that is constant over $\mathbb{R}^m$ and know σ^2. In this case, the posterior distribution on u_i at time t is multivariate normal with mean $u_{i,t}$, and the expected value of $\theta_t \cdot u_i$ under this posterior conditioned on θ_t is $\theta_t \cdot u_{i,t}$ (see equation 2.13 in section 2.5, **?**). Alternatively, one may simply take our assumption that agents use the average as their estimate of an attribute value directly without such a Bayesian justification.

We define the regret at time t as $r(t) = \max_i\{\theta_t \cdot u_i\} - \theta_t \cdot u_{A_t}$, and the cumulative regret up to time T as $R(T) = \sum_{t=1}^{T} r(t)$. Define the cumulative payment up to time T similarly as $C(t) = \sum_{t=1}^{T} c(t)$. As the principal, we want to find a strategy $\mathcal{A}$ under which both the cumulative expected regret $\mathbb{E}_{\mathcal{A}}[R(T)]$ and the cumulative expected payment $\mathbb{E}_{\mathcal{A}}[C(T)]$ are small.

To support later development, we define some additional notation. We let $B(\theta)$ and $\hat{B}(\theta)$ refer to the index of the arm that is best and second best for an

agent with preference vector θ, $B(\theta) \in \arg\max_i \theta \cdot u_i$ and $\hat{B}(\theta) = \arg\max_{i \neq B(\theta)} \theta \cdot u_i$, breaking ties uniformly at random. We let $N(i,t)$ denote the number of pulls of arm i at times up to and including t plus 1 (because of the initial pull), i.e. $N(i,t) = \sum_{s<t} 1\{A_s = i\} + 1$. We call time $t_n = \min_t\{\forall i, N(i,t) \geq n\}$ the *starting point of the n^{th} round*. We call the set of times $[t_n, t_{n+1})$ the *n^{th} round*.

5.3 Algorithm and Upper Bound

In this section, we propose a simple policy that mostly exploits, and occasionally incentivizes exploration when the probability of an arm would be pulled by all agent types below a time-varying threshold given the current posterior. We prove that with the help of heterogeneous preferences, we can get a certain amount of exploration for free via heterogeneity.

5.3.1 Our Algorithm

Our algorithm incentivizes pulling an arm i at a time t in round n if and only if both of the following criteria are met:

- the probability of pulling arm i would be below n^{-1} without incentives;
- arm i has not been played previously in the current round.

Ties are broken randomly. This algorithm does not need to know the horizon T in advance.

If our algorithm decides to incentivize an arm i, it uses the "pay whatever it takes" strategy in which the payment offered is $\max_{\theta,j} \theta \cdot (u_{j,t} - u_{i,t})$. This maximum over θ is taken over the support of F, which we recall is assumed compact. (We use this "pay whatever it takes" strategy for its simplicity, and in Section 5.3.4 we provide an alternate and smaller incentive payment that achieves the same payment budget bound and regret bound).

We describe our algorithm in detail as follows:

Algorithm 7 Algorithm: Incentivizing Exploration

Set n = 1 to denote the round number; Let $V = \emptyset$ be the set of arms that were pulled in the current round;
for $t = 1, 2, 3, \cdots$ **do**
  Let $S = \{i : P(\theta \cdot u_{i,t} > \theta \cdot u_{j,t} \; \forall j \neq i | u_{j,t} \; \forall j) < n^{-1}\}$ be the set of arms with unincentivized probability of being pulled below n^{-1}.
  if $S \setminus V$ is non-empty **then**
    Choose an arm i uniformly at random from $S \setminus V$
    Pay whatever it takes to incentivize pulling arm i, i.e., offer payment $c_{i,t} = \max_{\theta,j} \theta \cdot (u_{j,t} - u_{i,t})$ and $c_{j,t} = 0$ for $j \neq i$.
  else
    Let agents play myopically, i.e., offer payment $c_{j,t} = 0$ for all j
  end if
  Denote A_t as the pulled arm, update $V = V \cup \{A_t\}$, $u_{A_t,t}$ and $N(A_t, t)$
  if $n \neq \min_i N(i, t)$ **then**
    $V = \emptyset$
  end if
  Update the round number, $n = \min_i N(i, t)$
end for

5.3.2 Assumptions

In this section, we state several assumptions assumed by our analysis. First define

$$\Omega_{i,j} = \{\theta : B(\theta) = i, \hat{B}(\theta) = j\},$$

which is the set of agent preferences whose best arm is arm i and second best arm is arm j. With this definition, our analysis makes the following assumptions:

Assumption 1. Let $F_{i,j}(y)$ be the marginal cumulative density function (or cumulative mass function if $F(\cdot)$ is a discrete distribution) of $(u_i - u_j) \cdot \theta$ conditioned on $\theta \in \Omega_{i,j}$. We assume $F_{i,j}(y) \leq My$ for all $y \in R^+$, $\forall i, j$.

As we can see later in our proof, we only need $\max_{i,j} \limsup_{y \to 0^+} \frac{F_{i,j}(y)}{y}$ to be finite. Intuitively, assumption 1 states that there are not many agents who are indifferent between their best arm and the second best arm.

Assumption 2. We assume F has a compact support set. Without loss of generality, we assume $\theta \in [0, W]^m$.

We use $R = \max_{\theta,i,j} \theta \cdot (u_i - u_j)$ to denote the maximum regret that can be incurred at each time. Assumption 2 shows that $R < \infty$.

Assumption 3. Denote $p = \min_i P(\{\theta : B(\theta) = i\})$. We assume $p > 0$.

Assumption 3 means each arm i has a strictly positive proportion of users for which that arm is best.

5.3.3 General Results

In this section, we prove Algorithm 7 achieves $O(N^2 + M(\log(T))^2)$ cumulative regret with $O(N^2)$ payment budget. This is stated in the following pair of theorems, which together constitute our main results.

Theorem 7. The payment budget for Algorithm 7 is bounded above by $O(N^2)$.

Theorem 8. The cumulative regret for Algorithm 7 is bounded above by $O(N^2m + Mm^2(\log(T))^2)$.

Before we prove these two theorems, we must first introduce two additional pieces of notation, which will be used in preliminary lemmas. Let $S(\delta)$ be the proportion of users whose utility difference between their best and second best arm is less than δ. Formally, $S(\delta) = P(\theta : \theta \cdot u_{B(\theta)} - \theta \cdot u_{\hat{B}(\theta)} \leq \delta)$. Then, let $p(\delta) = \min_i P(\{\theta : B(\theta) = i, \theta \cdot u_{B(\theta)} - \theta \cdot u_{\hat{B}(\theta)} > \delta\})$. We know $p(0) = p$.

With this additional notation, we now prove several lemmas. First, based on Assumption 1, we have the following bound for $S(\delta)$.

Lemma 10. $S(\delta) \leq M\delta$.

Proof.

$$\begin{aligned} S(\delta) &= \sum_{i,j} P(\theta \cdot (u_i - u_j) \leq \delta | \theta \in \Omega_{i,j}) P(\theta \in \Omega_{i,j}) \\ &\leq \sum_{i,j} M\delta \times P(\theta \in \Omega_{i,j}) \\ &= M\delta. \end{aligned}$$

□

The following lemma bounds the probability of making a mistake if we let the agents play myopically in the n^{th} round, given that the utility difference between his/her best and second best arm is bounded below by a constant.

Lemma 11. Define τ to be any stopping time that is almost surely between t_n and $t_{n+1} - 1$ with respect to the filtration $\mathcal{F}_t = \sigma(A_1, \cdots, A_t, c_1, \cdots, c_t, O_1, \cdots, O_t)$,

we have

$$P(\arg\max\{\theta_\tau \cdot u_{i,\tau}\} \neq B(\theta_\tau) | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \leq 24Nm \exp\left(-\frac{1.8n\lambda^2}{16\sigma^2}\right),$$

for $n \geq n_0 = \max\{50, \frac{92.16\sigma^4}{\lambda^4}\}$.

We need the following lemma in order to prove Lemma 11.

Lemma 12. For $n \geq n_0 = \max\{50, \frac{92.16\sigma^4}{\lambda^4}\}$, we have

$$\frac{n\lambda}{4\sigma} \geq \sqrt{0.6n \log(\log_{1.1}(n) + 1)}.$$

Proof. First, we observe that

$$\begin{aligned} &\frac{n\lambda}{4\sigma} \geq \sqrt{0.6n \log(\log_{1.1}(n) + 1)} \\ \iff &\frac{n}{\log(\log_{1.1}(n) + 1)} \geq \frac{9.6\sigma^2}{\lambda^2}. \end{aligned}$$

Since $\log(x) \leq x - 1$ for $x > 0$, we know

$$\log(\log_{1.1}(n) + 1) = \log\left(\frac{\log(n)}{\log(1.1)} + 1\right) \leq \log(11\log(n) + 1) \leq \log(11n) \leq 3 + \log(n).$$

Thus, we know

$$\frac{n}{\log(\log_{1.1}(n) + 1)} \geq \frac{n}{3 + \log(n)}.$$

To prove the lemma, we just need to show for $n \geq n_0$, we have

$$\frac{n}{3 + \log(n)} \geq \frac{9.6\sigma^2}{\lambda^2}. \tag{5.1}$$

Inequality (5.1) is true because of the following two observations:

- for $n \geq 50$, we have $\frac{n}{3+\log(n)} \geq n^{0.5}$;
- for $n \geq \frac{92.16\sigma^4}{\lambda^4}$, we have $n^{0.5} \geq 9.6\sigma^2\lambda^2$.

Thus, we know our lemma is true.

□

To prove Lemma 11, we also need to use an adaptive concentration inequality due to Zhao et al. (2016). For reference, we state it here as a Lemma.

Lemma 13 (Corollary 1 in Zhao et al. (2016))**.** Let X_i be zero mean 1/2-subgaussian random variables. $\{S_n = \sum_{i=1}^n X_i, n \geq 1\}$ be a random walk. Let J be any stopping time with respect to $\{X_1, X_2, \cdots\}$. We allow J to take the value of ∞ where $P(J = \infty) = 1 - \lim_{n\to\infty} P(J \leq n)$. If

$$f(n) = \sqrt{0.6n \log(\log_{1.1}(n) + 1) + bn},$$

then

$$Pr[\{S_J \geq f(J)\} \cap \{J < \infty\}] \leq 12e^{-1.8b}.$$

We now prove Lemma 11.

Proof of Lemma 11. In the n^{th} round, we know all arms have been pulled at least n times. For all the agents θ whose utility difference between their best and second best arm is greater than $2mW\lambda$, denote $K(\theta) = \max_{i\neq B(\theta)}\{\theta \cdot u_{i,t}\}$. If $|u_{i,t}^j - u_i^j| \leq \lambda$ for

all i, j, then

$$\begin{aligned}
&\theta \cdot (u_{B(\theta),t} - u_{K(\theta),t}) \\
\geq &\theta \cdot (u_{B(\theta),t} - u_{B(\theta)}) + \theta \cdot (u_{K(\theta)} - u_{K(\theta),t}) + \theta \cdot (u_{B(\theta)} - u_{K(\theta)}) \\
> &- Wm\lambda - Wm\lambda + 2Wm\lambda = 0,
\end{aligned}$$

which means their myopic action would incur no regret.

Define $\epsilon_{i,\tau} = u_{i,\tau} - u_i$ and $\epsilon_{i,\tau}^j$ to be the j^{th} component of $\epsilon_{i,\tau}$. Thus, we have

$$\begin{aligned}
&P(\arg\max\{\theta_\tau \cdot u_{i,\tau}\} \neq B(\theta_\tau) | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \\
\leq &P(\exists i, \exists j, |u_{i,\tau}^j - u_i^j| \geq \lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \\
\leq &\sum_i \sum_j P(|u_{i,\tau}^j - u_i^j| \geq \lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \\
= &\sum_i \sum_j P(|\epsilon_{i,\tau}^j| \geq \lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda). \qquad (5.2)
\end{aligned}$$

To bound equation (5.2), we use Lemma 13. Define

$$S_{N(i,\tau)}^{i,j} = \frac{\epsilon_{i,\tau}^j}{2\sigma}.$$

Based on Lemma 12, for $n_0 = \max\{50, \frac{92.16\sigma^2}{\lambda^2}\}$ and $n \geq n_0$, we have

$$\frac{n\lambda}{4\sigma} \geq \sqrt{0.6n \log(\log_{1.1}(n) + 1)}.$$

Thus, if we set $b = \frac{n\lambda^2}{16\sigma^2}$ in Lemma 13, for any $N(i,\tau) \geq n \geq n_0$, we have

$$\begin{aligned}
\frac{N(i,\tau)\lambda}{2\sigma} &\geq \sqrt{0.6N(i,\tau) \log(\log_{1.1}(N(i,\tau)) + 1)} + \frac{\lambda}{4\sigma} \sqrt{nN(i,\tau)} \\
&\geq \sqrt{0.6N(i,\tau) \log(\log_{1.1}(N(i,\tau)) + 1) + bN(i,\tau)},
\end{aligned}$$

where the last inequality is because $\sqrt{x} + \sqrt{y} \geq \sqrt{x+y}$. Thus, we have

$$\begin{aligned}
&P(\epsilon_{i,\tau}^j \geq \lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \\
=&P\left(S_{N(i,\tau)}^{i,j} \geq \frac{N(i,\tau)\lambda}{2\sigma} \middle| \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda\right) \\
\leq&P\left(S_{N(i,\tau)}^{i,j} \geq \sqrt{0.6N_{i,\tau}\log(\log_{1.1}(N(i,\tau))+1) + bN(i,\tau)} \middle| \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda\right) \\
\leq&12\exp(-1.8b) = 12\exp\left(\frac{-1.8n\lambda^2}{16\sigma^2}\right).
\end{aligned}$$

Similarily, we can bound

$$\begin{aligned}
&P(\epsilon_{i,\tau}^j \leq -\lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \\
=&P(-\epsilon_{i,\tau}^j \geq \lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \\
\leq&12\exp\left(\frac{-1.8n\lambda^2}{16\sigma^2}\right).
\end{aligned}$$

Therefore, we know $P(|\epsilon_{i,\tau}^j| \geq \lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \leq 24\exp\left(\frac{-1.8n\lambda^2}{16\sigma^2}\right)$. Thus, we know

$$\begin{aligned}
&\sum_i \sum_j P(|\epsilon_{i,\tau}^j| \geq \lambda | \theta_\tau \cdot (u_{B(\theta_\tau)} - u_{\hat{B}(\theta_\tau)}) > 2Wm\lambda) \\
\leq&24Nm\exp\left(\frac{-1.8n\lambda^2}{16\sigma^2}\right).
\end{aligned}$$

□

Before we start analyzing the cumulative regret, we first prove the following lemma which bounds the expected length of each round.

Lemma 14. Using our algorithm, we have $\mathbb{E}[t_{n+1} - t_n] \leq Nn, \forall n \geq 1$.

Proof. A round completes when each arm is pulled at least once in that round. Let X_i be the number of agents who come to the system between the time after

the $(i-1)^{th}$ unique arm was pulled, up to and including the time when the i^{th} unique arm was pulled. Then we know

$$\mathbb{E}[t_{n+1} - t_n] = \sum_{i=1}^{N} E[X_i].$$

Fix i. In bounding X_i, we think of agents as "trials", where each trial can result in a new unique arm being pulled (which we call a "successful" trial), or not. There are two ways a trial can be successful:

- If there is at least one arm that has not been pulled and the probability of an agent utility function that would pull this arm without incentives is less than n^{-1}, then the principal will offer an incentive that causes this arm to be pulled (or one of these arms if there is more than one). In this case, the probability that the trial is succesful is 1.
- The probability of an agent utility function that would pull each un-pulled arm without incentives is at least n^{-1}. In this case, the probability that the trial is successful is at least n^{-1}.

Thus, X_i is stochastically dominated below by a geometric random variable with success probability n^{-1}, the expected number of trials up to and including the first success, $E[X_i]$, is bounded above by n. Thus,

$$E[t_{n+1} - t_n] \leq Nn.$$

□

We also need the following lemma in part of the proof of Theorem 7.

Lemma 15. For all $n \geq 1$, we have

$$0.9n^{5/6} \geq \sqrt{0.6n \log(\log_{1.1}(n) + 1)}.$$

Proof. Since

$$\begin{aligned} & 0.9n^{5/6} \geq \sqrt{0.6n\log(\log_{1.1}(n)+1)} \\ \iff & 0.81n^{5/3} \geq 0.6n\log(\log_{1.1}(n)+1) \\ \iff & \frac{81}{60}n^{2/3} \geq \log(\log_{1.1}(n)+1), \end{aligned} \tag{5.3}$$

we only need to show (5.3) is true.

Denote $f(x) = \frac{81}{60}x^{2/3} - \log(\log_{1.1}(x)+1)$. It's easy to compute $f^{'}(x) = 0$ has a unique solution $x_0 = e^{2/3w(\frac{20e^{20000/314763}}{27})-\frac{10000}{104921}}$ (here $w(\cdot)$ is the Lambert W-Function) and it is the global minimum. Since $f(x_0) \approx 0.0252 > 0$, we know $f(x) > 0$ for all $x \geq 1$. Thus, our lemma holds true. □

Now we are ready to prove our first main result in this Chapter, Theorem 7.

Proof. Denote $\epsilon_{i,t} = u_{i,t} - u_i$ to be the estimation error for the attribute vector u_i at time t. Denote $\epsilon_{i,t}^j$ to be the j^{th} component of $\epsilon_{i,t}$. Denote ω to be a sample path and $n(t,\omega)$ to be the round number for sample path ω at time t. For a fixed time t, define

$$L^{'}[l](t) = \{\omega : |\epsilon_{i,t}^j(\omega)| \leq g(n(t,\omega),l), \forall i,j\},$$

where $g(n,l)$ is a function which we will define later. Define $L[1](t) = L^{'}[1](t)$ and $L[i](t) = L^{'}[i](t) \setminus L^{'}[i-1](t)$ for $i \geq 2$. We call $L[l](t)$ the l^{th} envelope at time t. We often simplify the notation and use $L[l]$ instead of $L[l](t)$ without confusion.

In the calculation below, we omit the dependency on ω when refering to variables $c(t)$, $\epsilon_{i,t}^j$ and t_n. Based on the definition of $L[l]$, we know if $\omega \in L[l]$, the

maximum payment we need to offer at time t is bounded above by

$$\begin{aligned}
&\max_i \theta_t \cdot u_{i,t} - \min_j \theta_t \cdot u_{j,t} \\
=&\max_i \theta_t \cdot (\epsilon_{i,t} + u_i) - \min_j \theta_t \cdot (\epsilon_{j,t} + u_j) \\
\leq&\max_i \theta_t \cdot u_i - \min_j \theta_t \cdot u_j + \max_i \theta_t \cdot \epsilon_{i,t} - \min_j \theta_t \cdot \epsilon_{j,t} \\
\leq&R + 2Wmg(n,l).
\end{aligned}$$

Based on the above notations, we can rewrite the cumulative payment as follows:

$$\begin{aligned}
&\sum_{t=1}^{\infty} c(t) \\
=&\sum_{l=1}^{\infty}\sum_{t=1}^{\infty} c(t)\mathbb{1}\{\omega \in L[l]\} \\
=&\sum_{l=1}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\infty} c(t)\mathbb{1}\{\omega \in L[l]\}\mathbb{1}\{t \in [t_n, t_{n+1})\}.
\end{aligned}$$

Set $g(n,l)$ to be $\frac{2\sigma l}{n^{1/6}}$. Since if $|u_i^j - u_{i,t}^j| \leq \lambda$ is true $\forall i$, $\forall j$, then we know for those $\theta \in \{\theta : \theta \cdot u_{B(\theta)} - \max_{j \neq B(\theta)}\{\theta \cdot u_j\} > 2Wm\lambda\}$, they will correctly identify their best arm. Thus, if $|u_i^j - u_{i,t}^j| \leq \frac{2\sigma l}{n^{1/6}} \leq \frac{p^{-1}(\frac{p}{2})}{2Wm}$ $\forall i$ and $\forall j$, then the probability that an unincentivized agent would pull arm i is at least $\frac{p}{2}$. Further, if time t is in a round n that satisfies $n^{-1} \leq p/2$, then our algorithm will not incentivize pulling any arms. Denote $a_0 = \frac{4Wm\sigma}{p^{-1}(\frac{p}{2})}$. In order to have $\frac{2\sigma l}{n^{1/6}} \leq \frac{p^{-1}(\frac{p}{2})}{2Wm}$, it is sufficient to have $n \geq \lceil (a_0 l)^6 \rceil$. In order to have $n^{-1} \leq \frac{p}{2}$, we need $n \geq \frac{2}{p}$. Denote $n_2 = \frac{2}{p}$. Thus, we know we can only incur regret for sample paths ω in the l^{th} envelope in the first $\max\{n_2, \lceil (a_0 l)^6 \rceil\}$ rounds.

Thus,

$$\sum_{t=1}^{\infty} c(t) = \sum_{l=1}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2, \lceil (a_0 l)^6 \rceil\}} c(t)\mathbb{1}\{\omega \in L[l]\}\mathbb{1}\{t \in [t_n, t_{n+1})\}.$$

Therefore,

$$
\begin{aligned}
&E\left[\sum_{t=1}^{\infty} c(t)\right] \\
=&\sum_{l=1}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0 l)^6\rceil\}} E[c(t)|\omega \in L[l], t \in [t_n, t_{n+1}))P(\omega \in L[l], t \in [t_n, t_{n+1})) \\
\leq&\sum_{l=1}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0 l)^6\rceil\}} \left[R + 2Wm\frac{2\sigma l}{n^{1/6}}\right] P(\omega \in L[l]|t \in [t_n, t_{n+1}))P(t \in [t_n, t_{n+1})) \\
\leq&\sum_{l=1}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0 l)^6\rceil\}} [R + 4Wm\sigma l]\, P(\omega \in L[l]|t \in [t_n, t_{n+1}))P(t \in [t_n, t_{n+1})).
\end{aligned}
$$

We now bound $P(\omega \in L[l]|t \in [t_n, t_{n+1}))$ for $n \geq n_0$ and $l \geq 2$. Using the definition of $L[l]$ and union bound, we have

$$
\begin{aligned}
&P(\omega \in L[l]|t \in [t_n, t_{n+1})) \\
=&P(\omega \in L^{'}[l]|t \in [t_n, t_{n+1})) - P(\omega \in L^{'}[l-1]|t \in [t_n, t_{n+1})) \\
\leq&1 - P\left(|\epsilon_{i,t}^{j}| < \frac{2\sigma(l-1)}{n^{1/6}}, \forall i, j|t \in [t_n, t_{n+1})\right) \\
=&P\left(\exists i, j, s.t.|\epsilon_{i,t}^{j}| \geq \frac{2\sigma(l-1)}{n^{1/6}}|t \in [t_n, t_{n+1})\right) \\
\leq&\sum_{i,j} P\left(|\epsilon_{i,t}^{j}| \geq \frac{2\sigma(l-1)}{n^{1/6}}|t \in [t_n, t_{n+1})\right).
\end{aligned}
$$

Define $S_{i,t}^{j} = \frac{N(i,t)\epsilon_{i,t}^{j}}{2\sigma}$, then we know $S_{i,t}^{j}$ is a summation of $1/2$ gaussian random numbers (here we can think of t as a stopping time that stops at time t in order to apply Lemma 13). Therefore,

$$\sum_{i,j} P\left(|\epsilon_{i,t}^{j}| \geq \frac{2\sigma(l-1)}{n^{1/6}} \middle| t \in [t_n, t_{n+1})\right)$$
$$= \sum_{i,j} P\left(|S_{i,t}^{j}| \geq \frac{N(i,t)(l-1)}{n^{1/6}} \middle| t \in [t_n, t_{n+1})\right)$$
$$\leq \sum_{i,j} P(|S_{i,t}^{j}| \geq N(i,t)^{5/6}(l-1) | t \in [t_n, t_{n+1})).$$

Based on Lemma 15, we know

$$N(i,t)^{5/6}(l-1)$$
$$=0.9N(i,t)^{5/6} + N(i,t)^{5/6}(l-1.9)$$
$$\geq \sqrt{0.6N(i,t)\log(\log_{1.1}(N(i,t))+1)} + \sqrt{(l-1.9)^2 N(i,t)}$$
$$\geq \sqrt{0.6N(i,t)\log(\log_{1.1}(N(i,t))+1) + (l-1.9)^2 N(i,t)}.$$

Therefore, based on Lemma 13, we know

$$\sum_{i,j} P(|S_{i,t}^{j}| \geq N(i,t)^{5/6}(l-1) | t \in [t_n, t_{n+1}))$$
$$\leq \sum_{i,j} 24e^{-1.8(l-1.9)^2} = 24Nme^{-1.8(l-1.9)^2}.$$

Thus,

$$
\begin{aligned}
&E\left[\sum_{t=1}^{\infty} c(t)\right] \\
&\leq \sum_{l=1}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0 l)^6\rceil\}} [R+4Wm\sigma l]\, P(\omega\in L[l]|t\in[t_n,t_{n+1}))P(t\in[t_n,t_{n+1})) \\
&\leq \sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0)^6\rceil\}} [R+4Wm\sigma]\, P(\omega\in L[1]|t\in[t_n,t_{n+1}))P(t\in[t_n,t_{n+1})) \\
&+ \sum_{l=2}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0 l)^6\rceil\}} [R+4Wm\sigma l]\, P(\omega\in L[l]|t\in[t_n,t_{n+1}))P(t\in[t_n,t_{n+1})) \\
&\leq \sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0)^6\rceil\}} [R+4Wm\sigma]\, P(t\in[t_n,t_{n+1})) \\
&+ \sum_{l=2}^{\infty}\sum_{t=1}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0 l)^6\rceil\}} [R+4Wm\sigma l]\, 24Nme^{-1.8(l-1.9)^2} P(t\in[t_n,t_{n+1})) \\
&\leq \sum_{n=1}^{\max\{n_2,\lceil(a_0)^6\rceil\}} [R+4Wm\sigma]\, Nn + \sum_{l=2}^{\infty}\sum_{n=1}^{\max\{n_2,\lceil(a_0 l)^6\rceil\}} [R+4Wm\sigma l]\, 24Nme^{-1.8(l-1.9)^2} Nn \\
&\leq [R+4Wm\sigma]\, N(\max\{n_2,\lceil(a_0)^6\rceil\})^2 + \sum_{l=2}^{\infty} 24N^2 m[R+4Wml\sigma](\max\{n_2,\lceil(a_0 l)^6\rceil\})^2 e^{-1.8(l-1.9)^2} \\
&= O(N^2).
\end{aligned}
$$

□

Lemma 16. The expected number of payments for Algorithm 7 is bounded above by $O(N^2)$.

Proof. If $|u_i^j - u_{i,t}^j| \leq \lambda$ is true $\forall i$, $\forall j$, then we know for those $\theta \in \{\theta : \theta \cdot u_{B(\theta)} - \max_{j\neq B(\theta)}\{\theta \cdot u_j\} > 2mW\lambda\}$, they will correctly identify their best arm. Thus we know, in the n^{th} round, if $|u_i^j - u_{i,t}^j| \leq \frac{p^{-1}(\frac{p}{2})}{2Wm}$ $\forall i$ and $\forall j$, and $n^{-1} \leq p/2$, we do not need to incentivize any arms. In order to have $n^{-1} \leq \frac{p}{2}$, we need $n \geq \frac{2}{p}$. Denote $n_1 = \max\{n_0, \frac{2}{p}\}$. Denote $\delta_0 = p^{-1}(\frac{p}{2}) > 0$ (because of Assumption 1).

Define τ_n^i to be the first time we pull arm i in the n^{th} round. Then

$$\sum_{t=1}^{\infty} \mathbb{1}\{c(t) > 0\} = \sum_{n=1}^{\infty} \sum_{i=1}^{N} \mathbb{1}\{c(\tau_n^i) > 0\}.$$

The cumulative expected number of payments is bounded above by:

$$\begin{aligned}
&E\left[\sum_{t=1}^{\infty} \mathbb{1}\{c(t) > 0\}\right] \\
=&\sum_{n=1}^{\infty} \sum_{i=1}^{N} P(c(\tau_n^i) > 0) \\
\leq&\sum_{n=n_1}^{\infty} \sum_{i=1}^{N} P\left(\exists i, j : |u_i^j - u_{i,\tau_n^i}^j| > \frac{p^{-1}(\frac{p}{2})}{2Wm}\right) + \sum_{n=1}^{n_1} N \\
\leq&\sum_{n=n_1}^{\infty} \sum_{i=1}^{N} 24Nm \exp\left(\frac{-1.8n\delta_0^2}{64W^2m^2\sigma^2}\right) + \sum_{n=1}^{n_1} N \\
\leq&\sum_{n=n_1}^{\infty} 24Nm \exp\left(\frac{-1.8n\delta_0^2}{64W^2m^2\sigma^2}\right) \times N + \sum_{n=1}^{n_1} N \\
\leq&24N^2m \frac{1}{\exp(\frac{1.8\delta_0}{64W^2m^2\sigma^2}) - 1} + Nn_1,
\end{aligned}$$

Thus, we know the expected number of payments is bounded above by $O(N^2)$.

□

Now we are ready to prove our second main result in this Chapter, Theorem 8.

Proof. For regret incurred in the first n_0 round, it is bounded above by $\sum_{n=1}^{n_0} NRn$.

For regret incurred after the first n_0 round, it has two different components: the regret incurred when we let the agents play myopically and the regret incurred when we incentivize the agents. Using Lemma 16, the expected

regret incurred when we incentivize the agents is bounded above by: $\left[24N^2m\frac{1}{\exp(\frac{1.8\delta_0}{64W^2m^2\sigma^2})-1}+Nn_1\right]R$.

For the regret incurred when we let the agents play myopically at time $t \geq t_{n_0}$, it consists of the following two components:

- For those users whose utility difference between their best and the second best arm is greater than $f(t)$: we define a sequence of stopping time τ_n^k to be the k^{th} time period in the n^{th} round. For $k > t_{n+1} - t_n$, we define $\tau_n^k = \infty$. For $\tau_n^k = t$, the probability of these users making a mistake is bounded above by $24Nm\exp\left(-\frac{1.8nf(\tau_n^k)^2}{64W^2m^2\sigma^2}\right)$ and the expected regret is bounded above by $24Nm\exp\left(-\frac{1.8nf(\tau_n^k)^2}{64W^2m^2\sigma^2}\right)\times R$. We denote the regret incurred by these agents as $r_1(\tau_n^k)$. For $k > t_{n+1} - t_n$, we define $r_1(\tau_n^k) = 0$.
- For those user whose utility difference between their best and the second best arm is smaller than $f(t)$: this happens with probability $S(f(t))$ at each time and regret is bounded above by $S(f(t))\times f(t) = Mf(t)^2$. We denote the regret incurred by these agents as $r_2(t)$.

Thus, the cumulative expected regret incurred up to time T when we let the agent play myopically is bounded above by:

$$\begin{aligned}
&E\left[\sum_{t=1}^{T} r(t)\right] \\
=&E\left[\sum_{t=1}^{t_{n_0}} r(t) + \sum_{t=t_{n_0}}^{T}(r_1(t)+r_2(t))\right] \\
\leq&\sum_{n=1}^{n_0} NRn + E\left[\sum_{n=n_0}^{T}\sum_{t=t_n}^{t_{n+1}-1} r_1(t)\right] + E\left[\sum_{t=1}^{T} r_2(t)\right] \\
=&\sum_{n=1}^{n_0} NRn + E\left[\sum_{n=n_0}^{T}\sum_{k=1}^{\infty} r_1(\tau_n^k)\right] + E\left[\sum_{t=1}^{T} r_2(t)\right].
\end{aligned} \tag{5.4}$$

Since

$$
\begin{aligned}
&E\left[\sum_{n=n_0}^{T}\sum_{k=1}^{\infty} r_1(\tau_n^k)\right] \\
&= \sum_{n=n_0}^{T}\sum_{k=1}^{\infty} E[r_1(\tau_n^k)] \\
&= \sum_{n=n_0}^{T}\sum_{k=1}^{\infty} (E[r_1(\tau_n^k)|\tau_n^k < \infty] \times P(\tau_n^k < \infty) + E[r_1(\tau_n^k)|\tau_n^k = \infty] \times P(\tau_n^k = \infty)) \\
&= \sum_{n=n_0}^{T}\sum_{k=1}^{\infty} E[r_1(\tau_n^k)|\tau_n^k < \infty] \times P(\tau_n^k < \infty),
\end{aligned}
$$

we have

$$
\begin{aligned}
(5.4) &= \sum_{n=1}^{n_0} NRn + \sum_{n=n_0}^{T}\sum_{k=1}^{\infty} E[r_1(\tau_n^k)|\tau_n^k < \infty] \times P(\tau_n^k < \infty) + E\left[\sum_{t=1}^{T} r_2(t)\right] \\
&\leq \sum_{n=1}^{n_0} NRn + \sum_{n=n_0}^{T}\left[\sum_{k=1}^{\infty} 24Nm \exp\left(-\frac{1.8nf(\tau_n^k)^2}{64W^2m^2\sigma^2}\right) R \times P(\tau_n^k < \infty)\right] + \sum_{k=1}^{T} Mf(t)^2 \\
&\leq \sum_{n=1}^{n_0} NRn + \sum_{n=1}^{T} 24Nm \exp\left(-\frac{1.8nf(n)^2}{64W^2m^2\sigma^2}\right) R \times Nn + \sum_{t=1}^{T} Mf(t)^2. \qquad (5.5)
\end{aligned}
$$

Thus the cumulative regret at time T is bounded above by

$$
\begin{aligned}
&\sum_{n=1}^{n_0} NRn + \sum_{n=1}^{T} 24Nm \exp\left(-\frac{1.8nf(n)^2}{64W^2m^2\sigma^2}\right) \times R \times Nn + \sum_{t=1}^{T} Mf(t)^2 \\
&+24N^2m \frac{1}{e^{\frac{1.8\delta_0}{64W^2m^2\sigma^2}} - 1} R + N\left(\max\left\{n_0, \frac{2}{p}\right\}\right) R.
\end{aligned}
$$

For a fixed T, we only need to minimize the following two terms since all others are constant:

$$
\sum_{n=1}^{T} 24Nm \exp\left(-\frac{1.8nf(n)^2}{64W^2m^2\sigma^2}\right) \times R \times Nn + \sum_{t=1}^{T} Mf(t)^2. \qquad (5.6)
$$

If we set $f^2(t) = \frac{2\log(T)\times 64W^2m^2\sigma^2}{1.8t}$, then

$$\begin{aligned}
&\sum_{n=1}^{T} 24Nm\exp\left(-\frac{1.8nf(n)^2}{64W^2m^2\sigma^2}\right)\times R\times Nn + \sum_{t=1}^{T} Mf(t)^2\\
\leq &\sum_{n=1}^{T} 24N^2mnR\exp\left(-2\log(T)\right) + \frac{128W^2m^2\sigma^2M\log(T)}{1.8}\sum_{t=1}^{T}\frac{1}{n}\\
\leq &24N^2mR\frac{T(T-1)}{2T^2} + 71.12W^2m^2\sigma^2M\log(T)(\log(T)+1)\\
\leq &12N^2mR + 71.12W^2m^2\sigma^2M\log(T)(\log(T)+1).
\end{aligned}$$

Thus, the cumulative expected regret is bounded by $O(N^2m+Mm^2\log(T))$. □

Corollary 1. If $\exists\delta > 0$ such that $F_{i,j}(\delta) = 0$ for all i, j, then the cumulative expected regret is bounded by $O(N^2)$.

Proof. The proof of this corollary is similar to the proof of Theorem 8. If we set $f^2(t) = \frac{2\log(T)\times 64W^2m^2\sigma^2}{1.8t}$, then there exists a t_0 such that for $t > t_0$, $S(f(t)) = 0$. Thus, similar to equation (5.5), we know the cumulative expected regret when we let the agents play myopically is bounded above by

$$\sum_{n=1}^{n_0} NRn + \sum_{n=1}^{T} 24Nm\exp\left(-\frac{1.8nf(n)^2}{64W^2m^2\sigma^2}\right)\times R\times Nn + \sum_{t=1}^{t_0} Mf(t)^2.$$

Therefore, based on the same analysis of Theorem 8, we know the cumulative regret is bounded by $O(N^2)$. □

5.3.4 Practical Issues

In Algorithm 7, we use "pay whatever it takes" strategy when we decide to incentivize the agent. However, "pay whatever it takes" only shows up in the proof of Lemma 14. Without loss of generality, suppose we want to incentivize

arm i at time t at the n^{th} round. Based on the proof of Lemma 14, as long as we offer a payment $c_{i,t}$ such that arm i has at least n^{-1} probability being pulled at time t, our results still hold true. We could compute this $c_{i,t}$ dynamically based on $F(\cdot)$ as well as our current estimate $u_{i,t}$. Here is the revised algorithm which would work well in practice:

Algorithm 8 Algorithm: Incentivizing Exploration

Set n = 1 to denote the round number; Let $V = \emptyset$ be the set of arms that were pulled in the current round;
for $t = 1, 2, 3, \cdots$ **do**
  Let $S = \{i : P(\theta \cdot u_{i,t} > \theta \cdot u_{j,t} \; \forall j \neq i | u_{j,t} \; \forall j) < n^{-1}\}$ be the set of arms with unincentivized probability of being pulled below n^{-1}.
  if $S \setminus V$ is non-empty **then**
    Choose an arm i uniformly at random from $S \setminus V$
    Offer payment $c_{i,t} = \inf\{c : P(\theta \sim F : \theta \cdot u_{i,t} + c > max_j \theta \cdot u_{j,t}) > n^{-1}\}$
  else
    Let agents play myopically, i.e., offer payment $c_{j,t} = 0$ for all j
  end if
  Denote A_t as the pulled arm, update $V = V \cup \{A_t\}$, $u_{A_t,t}$ and $N(A_t, t)$
  if $n \neq \min_i N(i, t)$ **then**
    $V = \emptyset$
  end if
  Update the round number, $n = \min_i N(i, t)$
end for

The same proof would work and we can get the exact same results as Algorithm 7.

5.4 Lower Bound $\Omega(\log(T))$

In this section, we assume θ follows a continuous distribution $F(\cdot)$. We provide an example to show the best possible lower bound is $\Omega(\log(T))$ regardless of the incentivizing strategy.

Suppose we have two arms. Arm 1 has attribute vector $(0, 0)$ and arm 2 has attribute vector $(0, 1)$. We assume the users' preference are uniformly distributed on the unit circle. If the user knows the exact attribute vectors for both arms, then the users with preference on the bottom half circle will choose arm 1 and the users with preference on the top half circle will choose arm 2.

Consider the following algorithm: at each step, let the agents play myopically; however, they are going to see the noisy rewards for both arms.

To lower bound the regret, we assume that the agents already know the true attribute vector for arm 1. Without loss of generality, denote $u_{2,t} = (0, 1) + (z_{t,1}, z_{t,2}) = (0, 1) + (N(0, 1/t), N(0, 1/t))$ to be the estimate attribute vector for arm 2 (Without loss of generality, we assume the variance for the noise is 1).

Since $(z_{t,1}, z_{t,2})$ is symmetric around $(0, 0)$, we know

$$
\begin{aligned}
&E[r(t)] \\
=&E[r(t)|z_{t,1} > 0, z_{t,2} > 0] \times P(z_{t,1} > 0, z_{t,2} > 0) + E[r(t)|z_{t,1} > 0, z_{t,2} < 0] \times P(z_{t,1} > 0, z_{t,2} < 0) \\
&+ E[r(t)|z_{t,1} < 0, z_{t,2} > 0] \times P(z_{t,1} < 0, z_{n,2} > 0) + E[r(t)|z_{t,1} < 0, z_{t,2} < 0] \times P(z_{t,1} < 0, z_{t,2} < 0) \\
\geq&0.25 \times E[r(t)|z_{t,1} > 0, z_{t,2} > 0].
\end{aligned}
$$

Given $z_{t,1} > 0$ and $z_{t,2} > 0$, we know users whose preference vectors between $(-1, 0)$ and $\left(\frac{-1-z_{t,2}}{\sqrt{z_{t,1}^2+(1+z_{t,2})^2}}, \frac{z_{t,1}}{\sqrt{z_{t,1}^2+(1+z_{t,2})^2}}\right)$ as well as users whose preference vectors between $(1, 0)$ and $\left(\frac{1+z_{t,2}}{\sqrt{z_{t,1}^2+(1+z_{t,2})^2}}, \frac{-z_{t,1}}{\sqrt{z_{t,1}^2+(1+z_{t,2})^2}}\right)$ will make a mistake. The regret is the absolute value of the second coordinate of the user's preference vector. Thus, we know

$$
\begin{aligned}
&E[r(t)|z_{t,1}>0, z_{t,2}>0] \\
=&4\times 2\int_0^{\infty}\int_0^{\infty}\int_0^{\arctan\left(\frac{z_{t,1}}{1+z_{t,2}}\right)}\frac{\sin(\theta)}{2\pi}d(\theta)\frac{e^{-\frac{t\times z_{t,1}^2}{2}}\sqrt{t}}{\sqrt{2\pi}}d(z_{t,1})\frac{e^{-\frac{t\times z_{t,2}^2}{2}}\sqrt{t}}{\sqrt{2\pi}}d(z_{t,2}) \\
=&\frac{2}{\pi^2}\int_0^{\infty}\int_0^{\infty}t\times\left[1-\frac{1+z_{t,2}}{\sqrt{z_{t,1}^2+(1+z_{t,2})^2}}\right]e^{-\frac{t\times z_{t,1}^2}{2}}e^{-\frac{t\times z_{t,2}^2}{2}}d(z_{t,1})d(z_{t,2}) \\
=&\frac{2}{\pi^2}\int_0^{\infty}\int_0^{\infty}\left[1-\frac{\sqrt{t}+z_{t,2}}{\sqrt{z_{t,1}^2+(\sqrt{t}+z_{t,2})^2}}\right]e^{-\frac{z_{t,1}^2}{2}}e^{-\frac{z_{t,2}^2}{2}}d(z_{t,1})d(z_{t,2}).
\end{aligned}
$$

Below, we want to show

$$\lim_{t\to\infty}\frac{E[r(t)|z_{t,1}>0, z_{t,2}>0]}{t}=O(1),$$

and use the fact that $\sum_{n=1}^{T}\frac{1}{n}=O(\log(T))$ to show the regret is at least $\Omega(\log(T))$.

Denote $d(t)=t\left[1-\frac{\sqrt{t}+z_{t,2}}{\sqrt{z_{t,1}^2+(\sqrt{t}+z_{t,2})^2}}\right]$. Since

$$d^{'}(t)=\frac{-z_{t,1}^2(2z_{t,2}+3\sqrt{t})-2(z_{t,2}+\sqrt{t})^3}{2(z_{t,1}^2+(z_{t,2}+\sqrt{t})^2)^{3/2}}+1,$$

and $\lim_{t\to\infty}d^{'}(t)=2$, we know for t large enough, $d(t)$ is a increasing function in terms of t. Based on the Monotone Convergence Theorem, we have

$$
\begin{aligned}
&\lim_{t\to\infty}\frac{E[r(t)|z_{t,1}>0, z_{t,2}>0]}{t} \\
=&\frac{2}{\pi^2}\int_0^{\infty}\int_0^{\infty}\lim_{t\to\infty}\left[t\left[1-\frac{\sqrt{t}+z_{t,2}}{\sqrt{z_{t,1}^2+(\sqrt{t}+z_{t,2})^2}}\right]e^{-\frac{z_{t,1}^2}{2}}e^{-\frac{z_{t,2}^2}{2}}\right]d(z_{t,1})d(z_{t,2}).
\end{aligned}
$$

Since $\lim_{t\to\infty} d(t) = \frac{z_{t,1}^2}{2}$, we know

$$\begin{aligned}
&\lim_{t\to\infty} \frac{E[r(t)|z_{t,1} > 0, z_{t,2} > 0]}{t} \\
=&\frac{2}{\pi^2} \int_0^\infty \int_0^\infty \lim_{t\to\infty} \left[\frac{z_{t,1}^2}{2} e^{-\frac{z_{t,1}^2}{2}} e^{-\frac{z_{t,2}^2}{2}} \right] d(z_{t,1}) d(z_{t,2}) = \frac{1}{2\pi}.
\end{aligned}$$

Therefore, the cumulative expected regret is at least $\Omega(\log(T))$.

5.5 Conclusion

In this chapter, we study attribute feedback with application in review aggregators such as Yelp. We study the incentivizing exploration problem with heterogeneous user preferences, which generalizes the problem setting studied by Frazier et al. (2014) and Han et al. (2015). We propose a simple policy that mostly exploits and occasionally incentivizes exploration, which can achieve $O(N^2 + M(\log(T))^2)$ cumulative expected regret with $O(N^2)$ payment budget.

CHAPTER 6

CONCLUSION

In this thesis, we study preference learning with three different types of user feedback. With each type of user feedback, we focus on one application and provide an algorithm and a theoretical analysis bounding its regret:

- In Chapter 2, we study cardinal feedback with application in information filtering systems. We provide an instance-specific computational upper bound and a pair of new *Decompose-Then-Decide* heuristic policies, DTD-DP and DTD-UCB, which works well in practice when compared to existing benchmarks;
- In Chapter 3 and Chapter 4, we study ordinal feedback with application in personalized recommender systems under the dueling bandits setting. We proposed two algorithms WS and CTB, which outperform existing benchmarks and have finite cumulative expected weak regret;
- In Chapter 5, we study attribute feedback with application in online review aggregators under the incentivizing exploration setting. We proposed an algorithm which mostly exploits and occasionally incentivizes exploration. We prove our algorithm can achieve $O(N^2 + M(\log(T))^2)$ cumulative expected regret with $O(N^2)$ payment budget.

APPENDIX A

APPENDIX OF "DUELING BANDITS WITH WEAK REGRET"

A.1 Gambler's Ruin Lemma

In our analysis of WS-W, we will use results from a special case of the Gambler's ruin problem (Karlin, 1968), stated as follows: suppose a gambler has m dollars initially. In each of a sequence of rounds, he loses 1 dollar with probability $q \neq \frac{1}{2}$ and wins 1 dollar with probability $1 - q$. He stops playing when he has either $m + 1$ dollars or has no money left. We have the following result, with a proof available on Page 73 of Karlin (1968).

Lemma 17 (Gambler's Ruin Lemma)**.** In the gambler's ruin problem: (1) the probability that the gambler reaches $m + 1$ dollars before reaching 0 dollars is $q_m = \frac{\left(\frac{1-q}{q}\right)^m - 1}{\left(\frac{1-q}{q}\right)^{m+1} - 1}$; (2) the expected number of steps before the gambler stops playing is $\frac{m}{1-2q} - \frac{m+1}{1-2q} \frac{\left(\frac{1-q}{q}\right)^m - 1}{\left(\frac{1-q}{q}\right)^{m+1} - 1}$.

Observe that the conditional distribution of $T_{\ell,k}$ and the winner of iteration k round ℓ, given the two arms being pulled, is given by the result above for the Gambler's ruin problem. We leverage this in our proof.

A.2 Proof of Lemma 1

Proof. Suppose we are comparing arm i versus arm j in this iteration with $i > j$ and arm i is the incumbent. Then we know $C(t_{\ell,k} - 1, i) = (N-1)(\ell - 1) + k - 1$ and $C(t_{\ell,k} - 1, j) = -\ell + 1$. We will keep playing these two arms until $C(t_{\ell,k} + T_{\ell,k} - 1, i) =$

$(N-1)(\ell-1)+k$ or $C(t_{\ell,k}+T_{\ell,k}-1, j) = (N-1)(\ell-1)+k$. Further, since the winning probability of arm i over arm j is $p_{i,j}$ over this period, we know the dynamics of this iteration are the same as those of the Gambler's Ruin problem. Denote $E = C(t_{\ell,k}-1, i) - C(t_{\ell,k}-1, j) + 1 = Nl + k - N$. Then the expected length of time we spend in this iteration by Lemma 17 is

$$\begin{aligned}
&\frac{E}{1-2p_{i,j}} - \frac{E+1}{1-2p_{i,j}} \frac{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^E - 1}{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^{E+1} - 1} \\
\leq &\frac{E}{1-2p_{i,j}} \leq \frac{E}{2p-1}.
\end{aligned}$$

The proof of second statement is similar. Using the same notation but now supposing $p_{i,j} \geq p > \frac{1}{2}$, we have that the expected length of time we spend in this iteration is

$$\begin{aligned}
&\frac{E}{1-2p_{i,j}} - \frac{E+1}{1-2p_{i,j}} \frac{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^E - 1}{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^{E+1} - 1} \\
= &\frac{1}{2p_{i,j}-1} - \frac{E+1}{1-2p_{i,j}} \frac{p_{i,j}(1-p_{i,j})^E - (1-p_{i,j})^{E+1}}{(1-p_{i,j})^{n+1} - p_{i,j}^{E+1}} \\
\leq &\frac{1}{2p-1}.
\end{aligned}$$

□

A.3 Proof of Lemma 2

In this section, we prove Lemma 2 from the Chapter 3. This section is structured as follows: In section A.3.1, we provide two bounds for the incumbent's losing and winning probability; In section A.3.2, we consider a version of the problem in which better and worse incumbents have constant (but different) winning

probabilities and provide a upper bound for the number of worse incumbents in a round before a better incumbent loses ; In section A.3.3, we use the results from the previous subsection to bound the expected number of iterations with a worse incumbent in a single round before a better incumbent loses, starting from within a round; In section A.3.4, we prove a similar bound on the expected number of iterations with a worse incumbent in this and future rounds before a better incumbent loses, starting from the beginning of a round; In section A.3.5, we complete the proof of Lemma 2.

Throughout this section, we use a one to one correspondence between n and (ℓ, k) defined by $n = (\ell - 1)(N - 1) + k$, $0 \leq k \leq N - 1$ and $\ell = \lceil n/(N-1) \rceil$. We also denote $p^* = \frac{2p-1}{p}$.

A.3.1 Bounds on Win and Loss Probabilities

We first prove the following two lemmas, which give

- a lower bound for the probability that a worse incumbent loses an iteration;
- an upper bound for the probability that a better incumbent loses an iteration.

Lemma 18. In iteration k of round ℓ conditioned on the identities of the incumbent and the challenger, if the incumbent is worse than the challenger, then the incumbent loses the iteration with conditional probability at least $p^* = \frac{2p-1}{p}$.

Proof. Let i be the incumbent and j be the challenger, with $i > j$. $C(i, t_{\ell,k}) \geq 0$ and

$C(j, t_{\ell,k}) \leq 0$. Let $E = C(i, t_{\ell,k}) + |C(j, t_{\ell,k})| + 1$. The probability that arm i loses this iterations is the same as $1 - q_E$ in the Gambler's Ruin Lemma, Lemma 17, with $q = p_{i,j} < 0.5$. This probability is:

$$\begin{aligned}
1 - q_E &= 1 - \frac{\left(\frac{1-p_{j,i}}{p_{i,j}}\right)^E - 1}{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^{E+1} - 1} \\
&\geq \frac{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^{E+1} - \left(\frac{1-p_{i,j}}{p_{i,j}}\right)^E}{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^{E+1}} = \frac{1 - 2p_{i,j}}{1 - p_{i,j}} \\
&\geq \frac{2p - 1}{p}.
\end{aligned}$$

□

Lemma 19. In iteration k of round ℓ conditioned on the identities of the incumbent and the challenger, if the incumbent is better than the challenger, then the incumbent loses the iteration with conditional probability at most $\left(\frac{1-p}{p}\right)^E$, where $E = N(\ell - 1) + k$.

Proof. This proof is similar to the previous one. Suppose we are pulling arm i and j with $i < j$ and i is the incumbent. Then we know $C(t_{\ell,k} - 1, i) = (N - 1)(\ell - 1) + k - 1$ and $C(t_{\ell,k} - 1, j) = -\ell + 1$. The probability that arm i loses is equal to $1 - q_E$ from the gambler's ruin problem, where $E = (N-1)(\ell-1)+k-1+\ell-1 = N(\ell-1)+k$. We have

$$\begin{aligned}
1 - q_E &= 1 - \frac{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^E - 1}{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^{E+1} - 1} \\
&= \frac{\left(\frac{1-p_{i,j}}{p_{i,j}}\right)^E \left[1 - \frac{1-p}{p}\right]}{1 - \left(\frac{1-p_{i,j}}{p_{i,j}}\right)^{E+1}} \\
&\leq \left(\frac{1 - p_{i,j}}{p_{i,j}}\right)^E \leq \left(\frac{1-p}{p}\right)^E.
\end{aligned}$$

□

A.3.2 Definition and Upper Bound for $g(b, m)$

In this section, we define a function $g(b, m)$ as follows. First, we define $g(0, m) = 0$ for any m. We define $g(b, m)$ for other integers b, m satisfying $m > 0$ and $0 \leq b \leq m$ recursively, as follows:

$$\begin{aligned}
&g(b, m) \\
=&\frac{b}{m} + \sum_{b'=0}^{b-1} \frac{1}{m} p^* g(b', m-1) + \sum_{b'=b}^{m-1} \frac{1}{m} g(b, m-1) \\
&+ \sum_{b'=0}^{b-1} \frac{1}{m} (1 - p^*) g(b-1, m-1) \\
=&\frac{b}{m} + \sum_{b'=0}^{b-1} \frac{1}{m} p^* g(b', m-1) + \frac{m-b}{m} g(b, m-1) \\
&+ \frac{b}{m} (1 - p^*) g(b-1, m-1) \qquad \text{(A.1)}
\end{aligned}$$

Intuitively, $g(b, m)$ is the expected number of future iterations in which the incumbent is worse than the challenger, starting with m arms that have not dueled yet b of which are better than the incumbent, when we stop counting when we reach the end of the round or when an incumbent loses to a worse challenger, in a simplified problem in which worse incumbents beat better challengers with probability p^*. In our problem, this probability is not p^*, but is bounded below by this quantity, and in the next section we will show that $g(b, m)$ is an upper bound on an analogous quantity in our problem.

We prove the following result about g.

Lemma 20. For $0 \leq b \leq m \leq N - 1$, we have

$$g(b, m) = g(b, b) \leq \frac{\log(b) + 1}{p^*}.$$

Proof. Given the boundary conditions $g(0,m) = 0$ for all m, we know Equation (A.1) has a unique solution. In this proof,

- We first assume $g(b,m) = g(b,b)$ for all $b \leq m$ and solve for $g(b,m)$;
- Then we show that this $g(b,m)$ is indeed the solution for Equation (A.1), verifying that $g(b,m)$ is as claimed;
- Finally, we show $g(b,m) \leq \frac{\log(b)+1}{p^*}$.

First, we solve for $g(b,m)$ with the assumption that $g(b,m) = g(b,b)$ for $b \leq m$. Setting $m = b$ in Equation (A.1) provides

$$g(b,b) = 1 + \sum_{b'=0}^{b-1} \frac{p^* g(b',b)}{b} + (1-p^*)g(b-1,b-1). \tag{A.2}$$

Thus, we know

$$\begin{aligned}
&\sum_{b'=1}^{b-1} p^* g(b',b+1) \\
&= \sum_{b'=1}^{b-1} p^* g(b',b) \\
&= b\left[g(b,b) - 1 - (1-p^*)g(b-1,b-1)\right].
\end{aligned}$$

Therefore, Equation (A.2) becomes

$$\begin{aligned}
&g(b+1,b+1) \\
=&1 + \frac{b}{b+1}[g(b,b) - 1 - (1-p^*)g(b-1,b-1)] \\
&+ \frac{p^* g(b,b)}{b+1} + (1-p^* g(b,b).
\end{aligned}$$

Re-organizing the terms, we have

$$\begin{aligned}&g(b+1,b+1)-g(b,b)\\=&\frac{1}{b+1}+\frac{b}{b+1}(1-p^*)[g(b,b)-g(b-1,b-1)].\end{aligned}$$

Denote $F(b)=g(b,b)-g(b-1,b-1)$. We know $F(1)=1$. Thus, we have

$$\begin{aligned}F(b)=&\frac{1}{b}+\frac{b-1}{b}(1-p^*)F(b-1)\\=&\frac{1}{b}+\frac{1-p^*}{b}+\frac{b-2}{b}(1-p^*)^2F(b-2)\\=&\frac{1}{b}+\frac{1-p^*}{b}+\cdots\frac{(1-p^*)^{b-1}}{b}.\end{aligned}$$

Therefore,

$$\begin{aligned}g(b,b)&=\sum_{k=1}^{b}F(k)\\&=\sum_{k=1}^{b}\left[\frac{1}{k}+\frac{1-p^*}{k}+\cdots\frac{(1-p^*)^{k-1}}{k}\right].\end{aligned}$$

Thus, if $g(b,m)=g(b,b)$ for all $b\le m$, we know

$$g(b,m)=\sum_{k=1}^{b}\left[\frac{1}{k}+\frac{1-p^*}{k}+\cdots\frac{(1-p^*)^{k-1}}{k}\right].$$

Now we verify that this is the correct solution. We prove this by induction on b. For $b=1$, Equation (A.1) becomes

$$g(1,m)=\frac{1}{m}+\frac{m-1}{m}g(1,m-1).$$

Since $g(1,1)=1$, it is easy to check $g(1,2)=g(1,3)=\cdots=g(1,N-1)=1$.

Suppose this $g(b,m)=g(b,b)$ are true for all $b\le m$, $b\le k$. For $b=k+1$,

Equation (A.1) becomes

$$
\begin{aligned}
& g(k+1,m) \\
=& \frac{k+1}{m} + \sum_{b'=0}^{k} \frac{p^*}{m} g(b', m-1) + \frac{m-k-1}{m} g(k+1, m-1) \\
& + \frac{k+1}{m}(1-p^*) g(k, m-1) \\
=& \frac{k+1}{m} + \sum_{b'=0}^{k} \frac{p^*}{m} g(b', b') + \frac{m-k-1}{m} g(k+1, m-1) \\
& + \frac{k+1}{m}(1-p^*) g(k,k).
\end{aligned}
$$

To show $g(k+1,m)$ does not depend on m, we need to prove the following equation is true for $m = k+2, k+3, \cdots, N-1$.

$$
\begin{aligned}
& \frac{k+1}{m} + \sum_{b'=0}^{k} \frac{p^*}{m} g(b', b') + \frac{k+1}{m}(1-p^*) g(k,k) \\
=& \frac{k+1}{m} g(k+1, m-1) \\
\Longleftrightarrow\ & k+1 + \sum_{b'=0}^{k} p^* g(b', b') + (k+1)(1-p^*) g(k,k) \\
=& (k+1) g(k+1, m-1) \qquad \text{(A.3)}
\end{aligned}
$$

We first check Equation (A.3) when $m = k+2$. Starting from the left hand side, we have

$$
\begin{aligned}
& k+1 + \sum_{b'=0}^{k} g(b', b') + (k+1)(1-p^*) g(k,k) \\
=& k+1 + (k+1)[g(k+1, k+1) - 1 - (1-p^*) g(k,k)] \qquad \text{(A.4)} \\
& + (k+1)(1-p^*) g(k,k) \\
=& (k+1) g(k+1, k+1),
\end{aligned}
$$

which equals to the right hand side. Equation (A.4) follows from Equation (A.2) (Equation (A.2) holds because $g(b,m) = g(b,b)$ for all $b \leq k$).

Again, by induction, we know (A.3) is true for all $m = k+2, \cdots, N-1$ and thus we concludes our induction.

We have shown that $g(b,m) = g(b,b)$ for all $b \leq m$.

Finally, we prove $g(b,b) = g(b,m) \leq \frac{\log(b)+1}{p^*}$. This is because

$$\begin{aligned}
g(b,m) =& g(b,b) \\
=& \sum_{k=1}^{b} \left[\frac{1}{k} + \frac{1-p^*}{k} + \cdots \frac{(1-p^*)^{k-1}}{k} \right] \\
\leq& \sum_{k=1}^{b} \left[\frac{1}{k} + \frac{1-p^*}{k} + \cdots \frac{(1-p^*)^{b-1}}{k} \right] \\
=& \sum_{k=1}^{b} \frac{1}{k} \left[1 + (1-p^*) + \cdots + (1-p^*)^{b-1} \right] \\
\leq& \frac{\log(b)+1}{p^*},
\end{aligned}$$

which concludes our proof. □

A.3.3 Bound on the Number of Iterations in One Round with a Worse Incumbent, Starting from Within the Round

Let $B(n)$ denote an indicator function that equals 1 if we have a better incumbent at the n^{th} iteration. The definition of $B(n)$ is very similar to $B(\ell, k)$ except $B(\ell, k)$ tracks both round and iteration number. Similarly, we use $\bar{B}(n) = 1 - B(n)$ to denote an indicator function that equals 1 if we have a worse incumbent at the n^{th} iteration.

Let $h(i, n, \mathcal{A})$ be the expected number of iterations with an incumbent that is worse than the challenger, between iteration n and the first time that a better

incumbent loses to a challenger or the round ends, given that the incumbent arm at iteration n is i and $\mathcal{A}$ is the set of arms that have not yet previously dueled in the round. Formally, we define this quantity as:

$$h(i, n, \mathcal{A}) = \mathbb{E}\left[\sum_{n'=n}^{\sigma-1} B(n') | \mathcal{A}, i_n = i\right],$$

where

- Conditioning on $\mathcal{A}$ is understood to mean that we are conditoning on $C(n-1, j) = -\ell + 1 \; \forall \; j \notin \mathcal{A} \cup \{i_n\}$, and $C(n-1, j) = -\ell \; \forall \; j \in \mathcal{A}$, where $\ell = \lceil n/(N-1) \rceil$ is the round in which iteration n resides. In other words, it is understood to mean that $\mathcal{A}$ contains the set of arms that have not yet dueled in this round.
- $\sigma = \min\{n' > n : J(n') = 1, n' = N\lceil n/(N-1) \rceil\}$ where $J(n)$ is an indicator that equals 1 when a better incumbent loses at iteration n, i.e., σ is the first time that either a better incumbent loses or the round ends.

Lemma 21. For any i, ℓ, k and $\mathcal{A}$, we have

$$h(i, n, \mathcal{A}) \leq g(b, m) \leq \frac{\log(N) + 1}{p^*},$$

where $m = N - k$ and $b = |\{u \in \mathcal{A} : u < i\}|$.

Proof. Denote $q_{i,j}(n)$ as the probability that incumbent arm i will beat challenger j at time n. We first write a recursive expression for $h(i, n, \mathcal{A})$ that applies when n is not divisible by N:

$$\begin{aligned} h(i, n, \mathcal{A}) = \sum_{\{j \in \mathcal{A} : i > j\}} \left[1 + \frac{q_{i,j}(n)}{N-k} h(i, n+1, \mathcal{A} \cup \{j\}) + \frac{1 - q_{i,j}(n)}{N-k} h(j, n+1, \mathcal{A} \cup \{i\})\right] \\ + \sum_{\{j \in \mathcal{A} : i < j\}} \frac{q_{i,j}(n)}{N-k} h(i, n+1, \mathcal{A} \cup j). \end{aligned} \tag{A.5}$$

When n is divisible by $N-1$, the only allowed value of $\mathcal{A}$ is $\emptyset$ and $h(i,n,\emptyset)=0$.

We then prove the desired result via induction on the number of iterations in the round, i.e., on $n \pmod{N-1}$. When $n \pmod{N-1}=0$, we have $h(i,n,\emptyset)=0$, $b=0$, and $g(b,m)=0$. Thus the result holds in this case.

Then suppose the result holds for all n with a particular value of $n \pmod{N-1}$ and we show it holds for $n-1$.

Applying the induction hypothesis to the right-hand side of (**??**), we have

$$h(i,n,\mathcal{A}) \leq \sum_{\{j\in\mathcal{A}:i>j\}} \left[1+\frac{q_{i,j}(n)}{m}g(b_{i,j},m-1)+\frac{1-q_{i,j}(n)}{m}g(b_{j,j},m-1)\right] + \sum_{\{j\in\mathcal{A}:i<j\}} \frac{q_{i,j}(n)}{m}g(b_{i,j},m-1), \tag{A.6}$$

where $b_{u,j}=\#\{u^{'}\in\mathcal{A}\setminus\{j\}:u^{'}<u\}$.

Consider the summand in the first sum in (A.6), dropping the constants 1 and $\frac{1}{m}$,

$$q_{i,j}(n)g(b_{i,j},m-1)+(1-q_{i,j}(n))g(b_{j,j},m-1). \tag{A.7}$$

This is increasing in $q_{i,j}(n)$ when $i>j$ since $b_{i,j}>b_{j,j}$, and since $g(b,m)$ is increasing in b. Since i is an incumbent that is worse than the challenger when $i>j$, Lemma 18 shows that $q_{i,j}(n)\leq 1-p^*=1-\frac{2p-1}{p}$ in this situation. Thus, this summand is bounded above by $(1-p^*)g(b_{i,j},m-1)+p^*g(b_{j,j},m-1)$.

Substituting this into (A.6), along with the inequality $q_{i,j}(n)\leq 1$ in the last

term, we have

$$
\begin{aligned}
& h(i, n, \mathcal{A}) \\
\leq & \sum_{\{j \in \mathcal{A}: i > j\}} \left[1 + \frac{1 - p^*}{m} g(b_{i,j}, m-1) + \frac{p^*}{m} g(b_{j,j}, m-1) \right] \\
& + \sum_{\{j \in \mathcal{A}: i < j\}} \frac{1}{m} g(b_{i,j}, m-1) \\
= & \frac{b}{m} + \frac{b}{m}(1 - p^*) g(b-1, m-1) + \sum_{b'=0}^{b-1} \frac{p^*}{m} g(b', m-1) \\
& + \frac{m-b}{m} g(b, m-1) \\
= & g(b, m)
\end{aligned}
$$

In the second to last line we have used that $\{b_{i,j} : j \in \mathcal{A}, i > j\} = \{0, \ldots, b-1\}$ and $b_{i,j} = b-1$ when $i > j$; $b_{i,j} = b$ when $i < j$; and that the cardinality of $\{j \in \mathcal{A} : i > j\}$ and $\{j \in \mathcal{A} : i < j\}$ are b and $m - b$ respectively. In the last line we have used the recursive definition of $g(b, m)$ in terms of $g(\cdot, m-1)$.

This shows the first inequality in the statement of the lemma. The second inequality follows directly from Lemma 20. □

A.3.4 Bound on the Number of Iterations with a Worse Incumbent, Starting from a Round Beginning

Denote $f(i, \ell)$ to be the expected number of iterations with a worse incumbent in this and future rounds, stopping as soon as a better incumbent loses, giving that we have arm i as the incumbent at the start of round ℓ.

Lemma 22. For any i and ℓ, we have

$$f(i, \ell) \leq \frac{\log(N) + 1}{(p^*)^2}.$$

Proof. Let $U(i, \ell)$ denote the expected number of iterations in round ℓ with a worse incumbent before a better incumbent loses. We use $V(\ell)$ to denote an indicator which equals to 1 if a better incumbent does not lose in the round ℓ. Then for $i > 1$,

$$f(i, \ell) = U(i, \ell) + \mathbb{E}[f(Z(\ell), \ell + 1)V(\ell)|Z(\ell - 1) = i].$$

The first term is bounded by Lemma 21 by

$$U(i, \ell) \leq \frac{\log(N) + 1}{p^*},$$

for all i and ℓ.

For the second term, since $f(Z(\ell), \ell + 1) = 0$ when $Z(\ell) = 1$, we know the second term is bounded by

$$\begin{aligned}
&\mathbb{E}[f(Z(\ell), \ell + 1)V(\ell)|Z(\ell - 1) = i] \\
\leq &\mathbb{E}[f(Z(\ell), \ell + 1)|Z(\ell) \neq 1, V(\ell) = 1, Z(\ell - 1) = i] \\
&\times P(Z(\ell) \neq 1, V(\ell)|Z(\ell - 1) = i).
\end{aligned}$$

Let $s_j = P(Z(\ell) = j|Z(\ell) \neq 1, V(\ell), Z(\ell-1) = i)$ to be the probability distribution over the integers from 2 through N. Then we know

$$\begin{aligned}
&\mathbb{E}[f(Z(\ell), \ell + 1)|Z(\ell) \neq 1, V(\ell) = 1, Z(\ell - 1) = i] \\
= &\sum_{j=2}^{N} s_j f(j, \ell + 1) \\
\leq &\max_{j=2,,N} f(j, \ell + 1).
\end{aligned}$$

Further, since if arm 1 wins its first duel as a challenger (which happens with probability at least p^*), then either $Z(\ell) = 1$ (it wins all subsequent duel in the round) or $V(\ell) = 0$ (it loses a subsequent duel), we have $P(Z(\ell) \neq 1, V(\ell)|Z(\ell-1) = i) \leq 1 - p*$.

Thus, we know

$$f(i, \ell) \leq \frac{\log(N) + 1}{p^*} + (1 - p^*) \max_{j=2,\cdots,N} f(j, \ell + 1).$$

Let $f(\ell) = \max_{j=2,\cdots,N} f(j, \ell)$. Then,

$$f(\ell) \leq \frac{\log(N) + 1}{p^*} + (1 - p^*) f(\ell + 1).$$

Thus,

$$\begin{aligned} f(1) \leq & \frac{\log(N) + 1}{p^*} + (1 - p^*) f(2) \\ \leq & \frac{\log(N) + 1}{p^*} (1 + (1 - p^*) + (1 - p^*)^2 + \cdots) \\ = & \frac{\log(N) + 1}{(p^*)^2}. \end{aligned}$$

□

A.3.5 Completing the Proof of Lemma 2

With the lemmas in the preceding subsections established, we now complete the proof of Lemma 2.

Proof. Let $\tau_0 = 0$ and $\tau_k = \{n > \tau_{k-1} : J(n) = 1\}$. The expected number of iterations with a worse incumbent is

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{n=0}^{\infty} \bar{B}(n)\right] \\
=&\mathbb{E}\sum_{k=0}^{\infty} 1\{\tau_k < \infty\} \sum_{n=\tau_k}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n) \\
=&\sum_{k=0}^{\infty} P(\tau_k < \infty)\mathbb{E}\left[\sum_{n=\tau_k}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n)|\tau_k < \infty\right]
\end{aligned}
$$

where we have used Tonelli's theorem to exchange the expectation of an infinite sum of non-negative terms with an infinite sum of expectations of the same terms.

Conditioning on the history available at time τ_k, we have that the inner expectation can be written as,

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{n=\tau_k}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n)|\tau_k < \infty\right] \\
=&\mathbb{E}\left[\mathbb{E}\left[\sum_{n=\tau_k}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n)|H_{\tau_k}, \tau_k < \infty\right]|\tau_k < \infty\right],
\end{aligned}
$$

where H_n is the sigma algebra generated by $(C(i,s) : s < t_{\ell,k'}, i = 1,\ldots,N)$, where $\ell = n \pmod{N-1}$, $k' = \lceil n/(N-1)\rceil$, and H_{τ_k} is the filtration $(H_n : n)$ stopped at τ_k.

We further break this inner term $\mathbb{E}\left[\sum_{n=\tau_k}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n)|H_{\tau_k}, \tau_k < \infty\right]$ into two parts: the part that occurs during the round in which τ_k resides, and the part that occurs in future rounds. Let $\ell_k = \lceil \tau_k/(N-1)\rceil$. Then,

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{n=\tau_k}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n)|H_{\tau_k}, \tau_k < \infty\right] \\
=&\mathbb{E}\left[\sum_{n=\tau_k}^{\ell_k N} 1\{n < \tau_{k+1}\}\bar{B}(n)|H_{\tau_k}, \tau_k < \infty\right] + \mathbb{E}\left[\sum_{n=\ell_k N+1}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n)|H_{\tau_k}, \tau_k < \infty\right] \\
\leq&\frac{\log(N)+1}{p^*} + \frac{\log(N)+1}{(p^*)^2} \\
\leq&\frac{2(\log(N)+1)}{(p^*)^2}
\end{aligned}
$$

where the second to last inequality relies on Lemma 21 to show $\mathbb{E}\left[\sum_{n=\tau_k}^{\ell_k N} 1\{n < \tau_{k+1}\}\bar{B}(n)|H_{\tau_k}, \tau_k < \infty\right]$ is bounded above by $\frac{\log(N)+1}{p^*}$ and Lemma 22 to show $\mathbb{E}\left[\sum_{n=\ell_k N+1}^{\infty} 1\{n < \tau_{k+1}\}\bar{B}(n)|H_{\tau_k}, \tau_k < \infty\right]$ is bounded above by $\frac{\log(N)+1}{(p^*)^2}$.

Thus,

$$\mathbb{E}\left[\sum_{n=0}^{\infty} \bar{B}(n)\right] \leq \frac{2(\log(N)+1)}{(p^*)^2} \sum_{k=0}^{\infty} P(\tau_k < \infty).$$

Now we bound $P(\tau_k < \infty)$ for a fixed k. Based on Lemma 19, we know $J(n)$ is a Bernoulli random variable with success rate less than $\left(\frac{1-p}{p}\right)^n$ (this is because of Lemma 19 and $n = (N-1)(\ell-1)+k < E$), independent across n. Let Q_n denote a Bernoulli random variable with success rate $\left(\frac{1-p}{p}\right)^n$. Then we know:

$$\begin{aligned} P(\tau_k < \infty) &\leq P\left(\sum_{i=1}^{\infty} J(i) \geq k\right) \\ &\leq P\left(\sum_{i=1}^{\infty} Q_i \geq k\right). \end{aligned}$$

Let $W_m = \sum_{i=1}^{m} Q_i$, which follows a Poisson Bernoulli distribution, and let $W = \lim_{m\to\infty} W_m$. W follows a Poisson distribution with parameter $\sum_{i=1}^{\infty}\left(\frac{1-p}{p}\right)^i = \frac{1-p}{2p-1}$ (Theorem 4, Wang (1993)). Thus,

$$\begin{aligned} \mathbb{E}\left[\sum_{n=0}^{\infty} \bar{B}(n)\right] &\leq \frac{2(\log(N)+1)}{(p^*)^2} \sum_{k=0}^{\infty} P(W \geq k) \\ &= \frac{2p^2(1-p)}{(2p-1)^3}(\log(N)+1) \\ &\leq \frac{2p^2}{(2p-1)^3}(\log(N)+1) \end{aligned}$$

□

A.4 Proof of Lemma 3

Proof. It is easy to see that at the last iteration which has a worse incumbent, the better arm is always arm 1. Thus, we only consider $C(t, 1)$ in this proof. At the end of the ℓ^{th} round, if $C(t_{\ell+1} - 1, 1) < 0$, we know $C(t_{\ell+1} - 1, 1) = -\ell$.

Let us consider a simple random walk W(t) such that $W(t + 1) = W(t) + 1$ with probability $p > \frac{1}{2}$ and $W(t + 1) = W(t) - 1$ with probability $1 - p$ for $t \geq 1$. If we denote $p^*_\ell = P(\exists t_*, W(t_*) = -\ell)$ for $\ell > 0$, then it is easy to calculate that $p^*_\ell = \left(\frac{1-p}{p}\right)^\ell$.

Now let us consider $C(t, 1)$. If we pull arm 1 with some other arm i at time t, then $C(t, 1) = C(t - 1, 1) + 1$ happens with probability $p_{1,i} > p$ and $C(t, 1) = C(t - 1, 1) - 1$ with probability $1 - p_{1,i} < 1 - p$. If we do not pull arm 1 at time t, then $C(t, 1) = C(t - 1, 1)$ with probability 1.

Define $\tau_1 = 1$ and $\tau_k = \min_t\{t > \tau_{k-1}, C(t, 1) \neq C(\tau_{k-1}, 1)\}$, for $k = 1, 2, \cdots,$. Because τ_k is a non-decreasing right continuous stopping time, we know it is a valid random change of time (Barndorff-Nielsen & Shiryaev, 2015). Define $R(k)$ a new stochastic process where $R(k) = C(\tau_k, 1)$. Then we know at every time k, $R(k) = R(k - 1) + 1$ with probability greater or equal to p and $R(k) = R(k - 1) - 1$ with probability less than 1-p. Define $p_\ell = P(\exists t_*, R(t_*) = -\ell)$, then it is easy to prove $p_\ell \leq p^*_\ell = \left(\frac{1-p}{p}\right)^\ell$ using first step analysis and induction (we leave the proof as an exercise for the reader), which means $P(\exists t_*, C(t_*, 1) = -\ell) \leq \left(\frac{1-p}{p}\right)^\ell$. □

A.5 Proof of Lemma 4

Proof. To show the first claimed equation, we have:

$$\begin{aligned}&\mathbb{E}[B(\ell,k)T_{\ell,k}\bar{D}(\ell)]\\ =&\mathbb{E}[B(\ell,k)T_{\ell,k}|\bar{D}(\ell)=1]P(\bar{D}(\ell)=1).\end{aligned}\tag{A.8}$$

The first term $\mathbb{E}[B(\ell,k)T_{\ell,k}|\bar{D}(\ell) = 1]$ can be bounded by writing it as $\mathbb{E}[B(\ell,k)T_{\ell,k}|\bar{D}(\ell) = 1] = \mathbb{E}[\mathbb{E}[B(\ell,k)T_{\ell,k}|\bar{D}(\ell) = 1, A(\ell,k)]|\bar{D}(\ell) = 1]$, where $A(\ell,k)$ denotes the pair of arms being pulled in iteration k round ℓ.

We focus on the inner term $\mathbb{E}[B(\ell,k)T_{\ell,k}|\bar{D}(\ell) = 1, A(\ell,k)]$. $B(\ell,k)$ is observable given $A(\ell,k)$. If $B(\ell,k) = 0$ then this inner term is 0. If $B(\ell,k) = 1$ then this inner term is $\mathbb{E}[T_{\ell,k}|A(\ell,k)]$ (where we note that $T_{\ell,k}$ is conditionally independent of $\bar{D}(\ell)$ given $A(\ell,k)$) and is bounded above by $1/(2p-1)$ by Lemma 1. In both cases, the inner term is bounded above by $1/(2p-1)$, and we have that $\mathbb{E}[B(\ell,k)T_{\ell,k}|\bar{D}(\ell) = 1] \leq 1/(2p-1)$.

Thus, we have that (A.8) is bounded above by

$$\frac{1}{2p-1}P(\bar{D}(\ell)=1) \leq \frac{1}{2p-1}\left(\frac{1-p}{p}\right)^{\ell-1},$$

where the final inequality follows from Lemma 3 and the fact that $\bar{D}(\ell) = 1$ implies $L \geq \ell - 1$.

To show the second claimed equation, we use the same proof technique used for the first and get:

$$\mathbb{E}[B(\ell,k)T_{\ell,k}V(\ell,k)] \leq \frac{1}{2p-1}P(V(\ell,k)=1).$$

Now we just need to compute $P(V(\ell,k) = 1)$. Given $C(t_\ell - 1, 1) = (N-1)(\ell-1)$ at the beginning of round ℓ, it loses only if there exists a $t_0 \geq t_\ell$ and $C(1,t_0) = -\ell$.

Using the results from Lemma 3, we know $P(V(\ell,k)=1) \le \left(\frac{1-p}{p}\right)^{\ell}$. This completes the proof of the second claimed equation. □

A.6 Proof of Lemma 5

Proof. For the first inequality, we know

$$\begin{aligned}&\mathbb{E}\left[\sum_{k=1}^{N-1}\bar{B}(\ell,k)T_{\ell,k}\bar{D}(\ell)\right]\\ =&\sum_{k=1}^{N-1}\mathbb{E}\left[\mathbb{E}[\bar{B}(\ell,k)T_{\ell,k}|D(\ell)=0]\bar{D}(\ell)\right].\end{aligned} \tag{A.9}$$

Moreover,

$$\begin{aligned}&\mathbb{E}[\bar{B}(\ell,k)T_{\ell,k}|D(\ell)=0]\\ =&\mathbb{E}[T_{\ell,k}|B(\ell,k)=0,D(\ell)=0]P(B(\ell,k)=0|D(\ell)=0)\\ \le&\frac{N\ell}{2p-1}P(B(\ell,k)=0|D(\ell)=0),\end{aligned}$$

where the last equation follows from applying Lemma 1 and iterated conditional expectation. Thus, we know

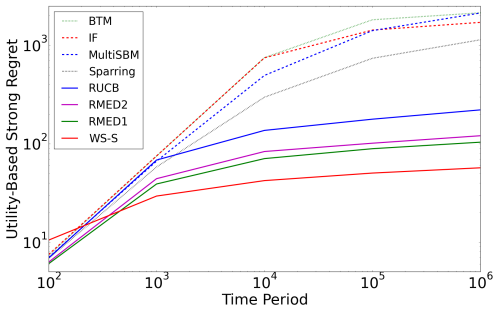
$$\begin{aligned}(A.9)&=\sum_{k=1}^{N-1}\frac{N\ell}{2p-1}P(B(\ell,k)=0|D(\ell)=0)\mathbb{E}[\bar{D}(\ell)]\\ &\le\sum_{k=1}^{N-1}\frac{N\ell}{2p-1}P(B(\ell,k)=0|D(\ell)=0)\left(\frac{1-p}{p}\right)^{\ell-1} \qquad (A.10)\\ &\le\left(\frac{1-p}{p}\right)^{\ell-1}\frac{2N\ell p^2}{(2p-1)^4}(\log(N)+1),\end{aligned}$$

where equation (A.10) is because Lemma 2.

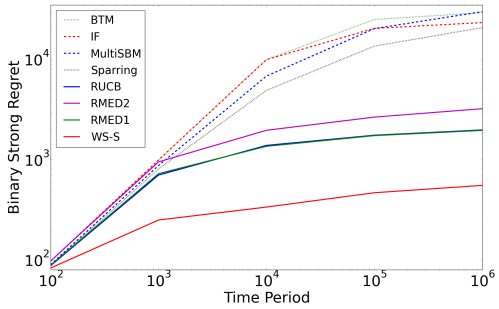
The proof of the second inequality follows very similarly, and is omitted. □

0.5	0.512	0.622	0.655	0.698	0.726	0.711	0.708	0.749	0.8	0.741	0.783	0.847	0.817	0.854	0.868
0.488	0.5	0.602	0.683	0.652	0.776	0.663	0.683	0.738	0.709	0.786	0.802	0.83	0.85	0.871	0.873
0.378	0.398	0.5	0.528	0.554	0.533	0.534	0.591	0.573	0.593	0.661	0.705	0.734	0.672	0.787	0.822
0.345	0.317	0.472	0.5	0.553	0.619	0.566	0.641	0.675	0.687	0.665	0.696	0.803	0.823	0.796	0.844
0.302	0.348	0.446	0.447	0.5	0.513	0.524	0.518	0.608	0.538	0.643	0.61	0.695	0.672	0.681	0.775
0.274	0.224	0.467	0.381	0.487	0.5	0.513	0.559	0.575	0.621	0.591	0.701	0.702	0.787	0.829	0.811
0.289	0.337	0.466	0.434	0.476	0.487	0.5	0.559	0.553	0.613	0.564	0.607	0.703	0.735	0.736	0.801
0.292	0.317	0.409	0.359	0.482	0.441	0.441	0.5	0.556	0.527	0.562	0.58	0.668	0.805	0.777	0.767
0.251	0.262	0.427	0.325	0.392	0.425	0.447	0.444	0.5	0.512	0.548	0.542	0.612	0.786	0.71	0.685
0.2	0.291	0.407	0.313	0.462	0.379	0.387	0.473	0.488	0.5	0.543	0.579	0.613	0.718	0.685	0.747
0.259	0.214	0.339	0.335	0.357	0.409	0.436	0.438	0.452	0.457	0.5	0.564	0.625	0.618	0.702	0.684
0.217	0.198	0.295	0.304	0.39	0.299	0.393	0.42	0.458	0.421	0.436	0.5	0.542	0.644	0.7	0.733
0.153	0.17	0.266	0.197	0.305	0.298	0.297	0.332	0.388	0.387	0.375	0.458	0.5	0.577	0.607	0.596
0.183	0.15	0.328	0.177	0.328	0.213	0.265	0.195	0.214	0.282	0.382	0.356	0.423	0.5	0.578	0.637
0.146	0.129	0.213	0.204	0.319	0.171	0.264	0.223	0.29	0.315	0.298	0.3	0.393	0.422	0.5	0.586
0.132	0.127	0.178	0.156	0.225	0.189	0.199	0.233	0.315	0.253	0.316	0.267	0.404	0.363	0.414	0.5

Figure A.1: User's preference matrix for the Sushi experiment



(a) Cyclic dataset with utility-based strong regret



(b) Cyclic dataset with binary strong regret

Figure A.2: Comparison of the strong regret between WS-S and 7 benchmarks on the cyclic dataset. WS-S outperforms all benchmarks in all settings studied.

A.7 Proof of Theorem 2

In this section, we prove the cumulative expected weak regret of WS-W is bounded by $O(N^2)$ in the Condorcet winner setting. First, we want to give an example to illustrate why our algorithm will not have $O(N\log(N))$ regret under the Condorcet winner setting.

In the Condorcet winner setting, Lemma 2 is no longer true. Here is a counter example to illustrate why Lemma 2 does not hold true anymore. Suppose we have $N = 3k+1$ arms in total, which includes a Condorcet winner arm and three types of other arms: k type-A arms, k type-B arms and k type-C arms. Among

these arms, we assume the user prefers type-A arms than type-B arms, type-B arms than type-C arms and type-C arms than type-A arms. Among each type of arms, there is a total order. In this setting, the expected number of iterations with a worse incumbent is $O(N)$ instead of $O(\log(N))$, which means Lemma 2 is no longer true.

Now we start our proof for Theorem 2.

Proof. In the Condorcent winner setting, Lemmas 3 and 4 hold, but as explained earlier, Lemma 2 does not. Because the proof of Lemma 5 utilizes Lemma 2, Lemma 5 also no longer holds.

On the other hand, since we can have at most $N-1$ iterations in a round, we know the following statement is true: the conditional expected number of iterations with a worse incumbent is bounded by N in each round. Thus, we know Lemma 5 now becomes:

$$\mathbb{E}\left[\sum_{k=1}^{N-1} \bar{B}(\ell,k) T_{\ell,k} \bar{D}(\ell)\right] \leq \left(\frac{1-p}{p}\right)^{\ell-1} \frac{N^2\ell}{2p-1},$$
$$\mathbb{E}\left[\sum_{k=1}^{N-1} \bar{B}(\ell,k) T_{\ell,k} V(\ell,k)\right] \leq \left(\frac{1-p}{p}\right)^{\ell} \frac{N^2\ell}{2p-1}.$$

Thus, following the same reasoning as in the proof of Theorem 1, we know the expected weak regret in the Condorcet winner setting is bounded by

$$\frac{NR}{(2p-1)^2} + \frac{pN^2}{(2p-1)^3},$$

which concludes our proof.

□

A.8 Preference Matrices

In the sushi experiment, the user's preference matrix is given by Figure A.1.

In the MSLR experiment, the ranker's preference matrix is given by:

$$\begin{bmatrix} 0.5 & 0.535 & 0.613 & 0.757 & 0.765 \\ 0.465 & 0.5 & 0.580 & 0.727 & 0.738 \\ 0.387 & 0.420 & 0.5 & 0.659 & 0.669 \\ 0.243 & 0.276 & 0.341 & 0.5 & 0.510 \\ 0.235 & 0.262 & 0.331 & 0.490 & 0.5 \end{bmatrix}$$

A.9 Condorcet Winner Experiment

In Chapter 3, we considered numerical examples in which the arms have a total order. This is common in the dueling bandits literature, where even work that considers more general settings theoretically test their methods on problems that satisfy the total order assumption (Komiyama et al., 2016; Urvoy et al., 2013).

In this section, we consider an additional example that has a Condorcet winner but does not have a total order among arms. The example has a cyclic struture, and is similar to the cyclic example in Komiyama et al. (2015).

The preference matrix is:

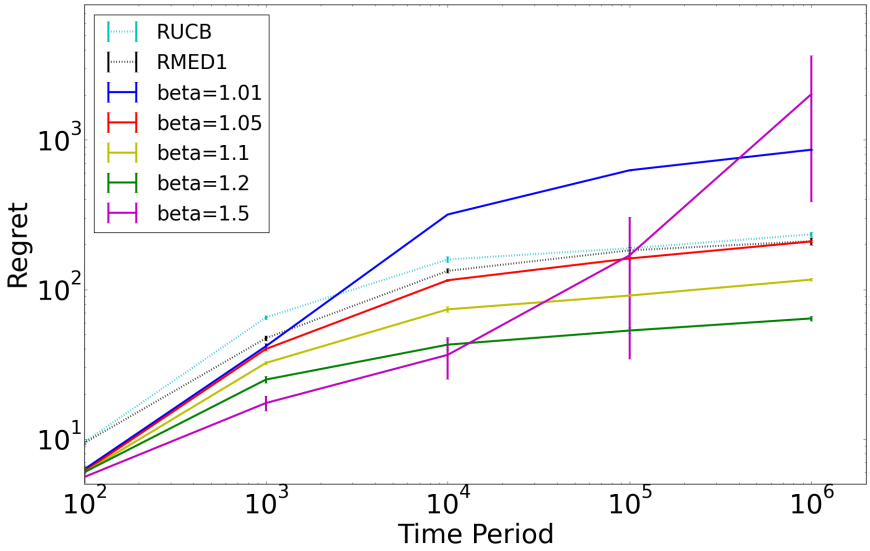


Figure A.3: Sensitivity Analysis

$$\begin{bmatrix} 0.5 & 0.6 & 0.6 & 0.6 \\ 0.4 & 0.5 & 0.6 & 0.4 \\ 0.4 & 0.4 & 0.5 & 0.6 \\ 0.4 & 0.6 & 0.4 & 0.5 \end{bmatrix}$$

In the above example, arm 1 is the Condorcet winner. Arm 2 beats arm 3, arm 3 beats arm 4 and arm 4 beats arm 2.

Again, we consider both binary strong regret and the utility-based strong regret. The utility-based strong regret is defined the same as the other two experiments. The result is summarized in Figure A.2. WS-S outperforms all benchmarks considered in all time periods on binary regret, and outperforms them all in all time periods except $T = 10^2$ on utility-based regret.

A.10 Sensitivity Analysis

In this section, we conduct a sensitivity analysis of β in WS-S using the MSLR dataset. In this analysis, we choose $\beta = 1.01, 1.05, 1.1, 1.2, 1.5$ respectively and compare them with RMED and RUCB. The result is summarized in Figure A.3.

Based on Figure A.3, WS-S with $\beta = 1.05, 1.1, 1.2$ outperforms RMED and RUCB. When $\beta = 1.01$, we spend too much time on the exploration period and do not exploit enough. Similarly, WS-S with $\beta = 1.5$ over exploits and does not explore enough. In both cases, WS-S underperforms RMED and RUCB. However, as long as β is within a reasonable range, WS-S can outperform existing state-of-art algorithms.

APPENDIX B

APPENDIX OF "DUELING BANDITS WITH DEPENDENT ARMS"

B.1 Proof of Lemma 6

First we prove another lemma.

Lemma 23. Suppose $Z(k)$ is a random walk starting with $Z(0) = 0$, $Z(k+1) = Z(k) + 1$ with probability $p > 0.5$ and $Z(k+1) = Z(k) - 1$ with probability $1 - p$. Then for $S \in \mathbb{N}$ we have

$$E\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq S\}\right] = \frac{p + S(2p-1)}{(2p-1)^2}. \tag{B.1}$$

Proof. Denote $A = \mathbb{E}[t : \min_{t>1} Z(t) = 0 | Z(1) = -1]$ and $B = P(\exists t, Z(t) = 0 | Z(1) = 1)$, then we know

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 0\}\right] = 1 + (1-p)\left(A + \mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 0\}\right]\right) + pB\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 0\}\right].$$

Now we need to calculate the expression for A and B respectively.

Based on the definition of A, we can rewrite A as $\mathbb{E}[t : \min_{t>1} Z(t) = 1 | Z(t) = 0]$. It is easy to show that $Y(t) := Z(t) - (2p-1)t$ is a martingale. Here we define a stopping time τ as $\min\{t > 1 : Z(1) = 1\}$. Then we know $Y(t)$ stops at τ is a martingale and thus $\mathbb{E}[Y(\tau)] = \mathbb{E}[Z(\tau)] - (2p-1)E[\tau] = 0$. Thus $A = \frac{1}{2p-1}$.

For B, based on the first step analysis, we know

$$B = (1-p) + p \times B^2.$$

Solving this equation, we get $B = \frac{1-p}{p}$.

Plus in A and B's expression, we have

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 0\}\right] = \frac{p}{(2p-1)^2}.$$

Now we compute $\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 1\}\right]$. Based on the same reasoning, we know

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 1\}\right] = 1 + (1-p)\left(A + \mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 1\}\right]\right) + p \times \mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 0\}\right].$$

Solving it, we get $\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq 1\}\right] = \frac{p+(2p-1)}{(2p-1)^2}$. For general S, we have

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq S\}\right] = 1 + (1-p)\left(A + \mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq S\}\right]\right) + p \times \mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq S-1\}\right],$$

by induction, we know our Lemma is true. □

Now we return to the proof of Lemma 1.

Proof. Suppose W(t) is a random walk and $W(t+1) = W(t) + 1$ with probability p and $W(t+1) = W(t) - 1$ with probability 1-p. Based on the previous Lemma, we just need to show

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{Z(t) \leq S\}\right] \leq \mathbb{E}\left[\sum_{t=0}^{\infty} \mathbb{1}\{W(t) \leq S\}\right]. \tag{B.2}$$

Because $E[\sum_{t=0}^{\infty} \mathbb{1}\{W(t) \leq S\}] = \sum_{t=0}^{\infty} P(W(t) \leq S)$ and

$$P(W(t) \leq S) = \sum_{2m \geq t-S} \binom{t}{m} p^{t-m}(1-p)^m \geq P(Z(t) \leq S),$$

we know Equation B.2 holds true. □

B.2 Proof of Lemma 8

Proof. We first prove it for $Y_t = 0$ and $x \in H_{i,j}$. This is because

$$\begin{aligned} p_{t+1}(x) &= p_{t+1}(\theta \in x) \\ &= P(\theta \in x | Y_t = 0, p_t(\cdot)) \\ &= \frac{P(\theta \in x, Y_t = 0, p_t(\cdot))}{P(Y_t = 0, p_t(\cdot))} \\ &= \frac{P(\theta \in x, Y_t = 0, p_t(\cdot))}{P(Y_t = 0, p_t(\cdot)|\theta \in H_{i,j})P(\theta \in H_{i,j}) + P(Y_t = 0, p_t(\cdot)|\theta \notin H_{i,j})P(\theta \notin H_{i,j})} \\ &= \frac{p_t(x)q}{p_t(H_{i,j})q + (1 - p_t(H_{i,j}))(1-q)}. \end{aligned}$$

The other three cases follow the same reasoning and we omit the proof. □

B.3 Proof of Lemma 9

Proof. We prove this lemma using induction. This is obviously true when t=0. Suppose this is true at time t-1. Without loss of generality, we write

$$p_{t-1}(C_k) = \frac{p_0(C_i)q^{m_i(t-1)-m_i(0)}(1-q)^{t-1-m_i(t-1)+m_i(0)}}{M(t-1)},$$

where $M(t-1)$ is a scaling constant. At time t, suppose we choose A_i and A_j for comparison and A_i wins the duel. Denote $M(t) = M(t-1) * [p_{t-1}(H_i j) * q + (1 - p_{t-1}(H_{i,j}))(1-q)]$, then if $C_k \in H_{i,j}$:

$$\begin{aligned} p_t(C_k) &= \frac{p_{t-1}(C_k)q}{p_{t-1}(H_{i,j})q + (1 - p_{t-1}(H_{i,j}))(1-q)} \\ &= \frac{p_0(C_k)q^{m_k(t-1)-m_k(0)}(1-q)^{t-1-m_k(t-1)+m_k(0)}q}{M(t-1)[p_{t-1}(H_{i,j})q + (1 - p_{t-1}(H_{i,j}))(1-q)]} \\ &= \frac{p_0(C_k)q^{m_k(t)-m_k(0)}(1-q)^{t-m_k(t)+m_k(0)}}{M(t)}, \end{aligned}$$

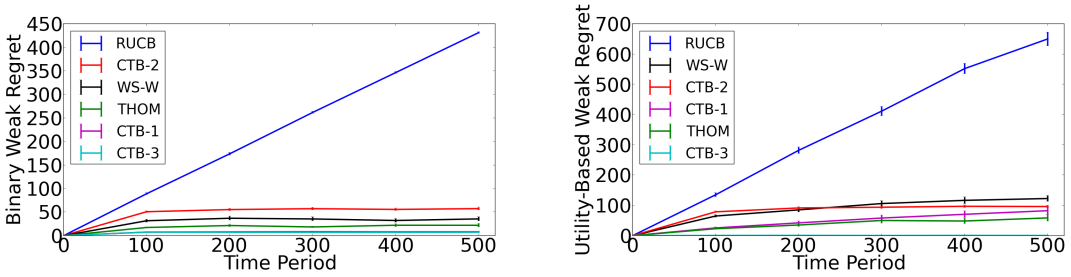
where the last line is based on the definition of $m_k(t)$ and $M(t)$. Similarly, if $C_k \notin H_i j$, then

$$\begin{aligned} p_t(C_k) &= \frac{p_{t-1}(C_k)(1-q)}{p_{t-1}(H_{i,j})q + (1-p_{t-1}(H_{i,j}))(1-q)} \\ &= \frac{p_0(C_k)q^{m_k(t-1)-m_k(0)}(1-q)^{t-1-m_k(t-1)+m_k(0)}(1-q)}{M(t-1)[p_{t-1}(H_{i,j})q + (1-p_{t-1}(H_{i,j}))(1-q)]} \\ &= \frac{p_0(C_k)q^{m_k(t)-m_k(0)}(1-q)^{t-m_k(t)+m_k(0)}}{M(t)}. \end{aligned}$$

□

B.4 Full Plot of Section 4.7

We include a plot which contains full information for RUCB. See Figure B.1 for details.



(a) Binary Regret and Constant $p_{i,j}$ (b) Bradley-Terry Regret and $p_{i,j}$

Figure B.1: Performance comparison of CTB−1, CTB−2, CTB−3, WS, RUCB and Thompson Sampling in the same experimental settings as in section 4.7, but with plots containing full information for RUCB.

BIBLIOGRAPHY

Abbasi-Yadkori, Yasin, Pál, Dávid, and Szepesvári, Csaba. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 2312–2320, 2011.

Agarwal, Deepak, Chen, Bee-Chung, and Pang, Bo. Personalized recommendation of user comments via factor models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 571–582. Association for Computational Linguistics, 2011a.

Agarwal, Deepak, Zhang, Liang, and Mazumder, Rahul. Modeling item-item similarities for personalized recommendations on yahoo! front page. *The Annals of Applied Statistics*, pp. 1839–1875, 2011b.

Agrawal, Shipra and Goyal, Navin. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pp. 127–135, 2013.

Ailon, Nir, Karnin, Zohar Shay, and Joachims, Thorsten. Reducing dueling bandits to cardinal bandits. In *ICML*, volume 32, pp. 856–864, 2014.

arXiv.org. arxiv.org. URL `http://arxiv.org/`.

Auer, Peter. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

Barndorff-Nielsen, Ole E and Shiryaev, Albert. *Change of time and change of measure*, volume 21. World Scientific Publishing Co Inc, 2015.

Blei, David M and Lafferty, John D. Topic models. *Text mining: classification, clustering, and applications*, 10(71):34, 2009.

Blei, David M, Ng, Andrew Y, and Jordan, Michael I. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

Brown, David B, Smith, James E, and Sun, Peng. Information relaxations and duality in stochastic dynamic programs. *Operations research*, 58(4-part-1): 785–801, 2010.

Busa-Fekete, Róbert and Hüllermeier, Eyke. A survey of preference-based online learning with bandit algorithms. In *International Conference on Algorithmic Learning Theory*, pp. 18–39. Springer, 2014.

Busa-Fekete, Róbert, Szörényi, Balázs, Cheng, Weiwei, Weng, Paul, and Hüllermeier, Eyke. Top-k selection based on adaptive sampling of noisy preferences. In *ICML (3)*, pp. 1094–1102, 2013.

Busa-Fekete, Róbert, Hüllermeier, Eyke, and Szörényi, Balázs. Preference-based rank elicitation using statistical models: The case of mallows. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1071–1079, 2014.

Cesa-Bianchi, Nicolo and Kakade, Sham. An optimal algorithm for linear bandits. *arXiv preprint arXiv:1110.4322*, 2011.

Chen, Bangrui and Frazier, Peter I. Dueling bandits with weak regret. *arXiv preprint arXiv:1706.04304*, 2017.

Chick, Stephen E and Frazier, Peter. Sequential sampling with economics of selection procedures. *Management Science*, 58(3):550–569, 2012.

Chu, Wei, Li, Lihong, Reyzin, Lev, and Schapire, Robert E. Contextual bandits with linear payoff functions. In *International Conference on Artificial Intelligence and Statistics*, pp. 208–214, 2011.

Dani, Varsha, Hayes, Thomas P, and Kakade, Sham M. Stochastic linear optimization under bandit feedback. In *COLT*, pp. 355–366, 2008.

Foltz, Peter W and Dumais, Susan T. Personalized information delivery: An analysis of information filtering methods. *Communications of the ACM*, 35(12): 51–60, 1992.

Franses, Philip Hans and Montgomery, Alan L. *Econometric models in marketing*, volume 16. Elsevier, 2002.

Frazier, Peter, Kempe, David, Kleinberg, Jon, and Kleinberg, Robert. Incentivizing exploration. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pp. 5–22. ACM, 2014.

Fürnkranz, Johannes and Hüllermeier, Eyke. Preference learning: An introduction. In *Preference learning*, pp. 1–17. Springer, 2010.

Gelman, Andrew, Carlin, John B, Stern, Hal S, Dunson, David B, Vehtari, Aki, and Rubin, Donald B. *Bayesian data analysis*, volume 2. CRC press Boca Raton, FL, 2014.

Han, Li, Kempe, David, and Qiang, Ruixin. Incentivizing exploration with heterogeneous value of money. In *International Conference on Web and Internet Economics*, pp. 370–383. Springer, 2015.

Hofmann, Katja, Whiteson, Shimon, and Rijke, Maarten De. Fidelity, soundness, and efficiency of interleaved comparison methods. *ACM Transactions on Information Systems (TOIS)*, 31(4):17, 2013.

Jamieson, Kevin G and Nowak, Robert. Active ranking using pairwise comparisons. In *Advances in Neural Information Processing Systems*, pp. 2240–2248, 2011.

Joachims, Thorsten, Granka, Laura, Pan, Bing, Hembrooke, Helene, Radlinski, Filip, and Gay, Geri. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Transactions on Information Systems (TOIS)*, 25(2):7, 2007.

Kamishima, Toshihiro. Nantonac collaborative filtering: recommendation based on order responses. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 583–588. ACM, 2003.

Karlin, Samuel. *A First Course In Stochastic Processes*. Academic Press, 1968.

Komiyama, Junpei, Honda, Junya, Kashima, Hisashi, and Nakagawa, Hiroshi. Regret lower bound and optimal algorithm in dueling bandit problem. In *COLT*, pp. 1141–1154, 2015.

Komiyama, Junpei, Honda, Junya, and Nakagawa, Hiroshi. Copeland dueling bandit problem: Regret lower bound, optimal algorithm, and computationally efficient algorithm. *arXiv preprint arXiv:1605.01677*, 2016.

Kremer, Ilan, Mansour, Yishay, and Perry, Motty. Implementing the wisdom of the crowd. *Journal of Political Economy*, 122(5):988–1012, 2014.

Lovejoy, William S. A survey of algorithmic methods for partially observed markov decision processes. *Annals of Operations Research*, 28(1):47–65, 1991.

Mansour, Yishay, Slivkins, Aleksandrs, and Syrgkanis, Vasilis. Bayesian incentive-compatible bandit exploration. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pp. 565–582. ACM, 2015.

Mansour, Yishay, Slivkins, Aleksandrs, Syrgkanis, Vasilis, and Wu,

Zhiwei Steven. Bayesian exploration: Incentivizing exploration in bayesian games. *arXiv preprint arXiv:1602.07570*, 2016.

May, Benedict C, Korda, Nathan, Lee, Anthony, and Leslie, David S. Optimistic bayesian sampling in contextual-bandit problems. *Journal of Machine Learning Research*, 13(Jun):2069–2106, 2012.

Monahan, George E. State of the arta survey of partially observable markov decision processes: theory, models, and algorithms. *Management Science*, 28 (1):1–16, 1982.

Pallone, Stephen N, Frazier, Peter I, and Henderson, Shane G. Bayes-optimal entropy pursuit for active choice-based preference learning. *arXiv preprint arXiv:1702.07694*, 2017.

Radlinski, Filip, Kurup, Madhu, and Joachims, Thorsten. How does clickthrough data reflect retrieval quality? In *Proceedings of the 17th ACM conference on Information and knowledge management*, pp. 43–52. ACM, 2008.

Rehurek, Radim and Sojka, Petr. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer, 2010.

Revelt, David and Train, Kenneth. Mixed logit with repeated choices: households' choices of appliance efficiency level. *Review of economics and statistics*, 80(4):647–657, 1998.

Robbins, Herbert. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Robbins, Herbert. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pp. 169–177. Springer, 1985.

Rusmevichientong, Paat and Tsitsiklis, John N. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

Russo, Dan and Van Roy, Benjamin. Learning to optimize via information-directed sampling. In *Advances in Neural Information Processing Systems*, pp. 1583–1591, 2014.

Schein, Andrew I, Popescul, Alexandrin, Ungar, Lyle H, and Pennock, David M. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 253–260. ACM, 2002.

Sullivan, Brian L, Wood, Christopher L, Iliff, Marshall J, Bonney, Rick E, Fink, Daniel, and Kelling, Steve. ebird: A citizen-based bird observation network in the biological sciences. *Biological Conservation*, 142(10):2282–2292, 2009.

Sutton, Richard S and Barto, Andrew G. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Urvoy, Tanguy, Clerot, Fabrice, Féraud, Raphael, and Naamane, Sami. Generic exploration and k-armed voting bandits. In *ICML (2)*, pp. 91–99, 2013.

Wang, Y.H. On the number of success in independent trials. *Statistica Sinica*, 1993.

Yelp, Inc. Yelp academic dataset, 2012. URL `https://www.yelp.com/dataset_challenge`.

Yue, Yisong and Joachims, Thorsten. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 1201–1208. ACM, 2009.

Yue, Yisong and Joachims, Thorsten. Beat the mean bandit. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 241–248, 2011.

Yue, Yisong, Broder, Josef, Kleinberg, Robert, and Joachims, Thorsten. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

Zhang, Yi, Xu, Wei, and Callan, James P. Exploration and exploitation in adaptive filtering based on bayesian active learning. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 896–903, 2003.

Zhao, Shengjia, Zhou, Enze, Sabharwal, Ashish, and Ermon, Stefano. Adaptive concentration inequalities for sequential decision problems. In *Advances In Neural Information Processing Systems*, pp. 1343–1351, 2016.

Zhao, Xiaoting and Frazier, Peter I. Exploration vs. exploitation in the information filtering problem. *arXiv preprint arXiv:1407.8186*, 2014.

Zoghi, Masrour, Whiteson, Shimon, Munos, Remi, Rijke, Maarten de, et al. Relative upper confidence bound for the k-armed dueling bandit problem. In *JMLR Workshop and Conference Proceedings*, number 32, pp. 10–18. JMLR, 2014.

Zoghi, Masrour, Karnin, Zohar S, Whiteson, Shimon, and De Rijke, Maarten. Copeland dueling bandits. In *Advances in Neural Information Processing Systems*, pp. 307–315, 2015.